캡스톤 디자인

‘딥페이크 탐지’

#9. Adversarial training II

김지수, 김민지, 민지민

지난 캡스톤 회의 내용

- **지난번 adversarial training 의 잘못된 점을 깨달아 다시 실험 진행했음**
- **샘플 논문 structure 구조 (대제목, 소제목 단락 구성)**
- **캡스톤 계획 발표 준비하기**

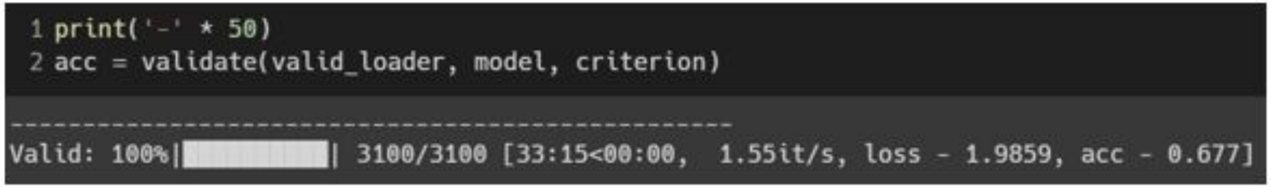
Gaussian noise test

생성한 노이즈 데이터셋을 xception 모델로 성능 측정

strong	**loss : 0.6845, acc: 0.680**
medium	**loss : 0.6944, acc : 0.325**
weak	**loss : 5.6372, acc : 0.323**

Salt and pepper noise test

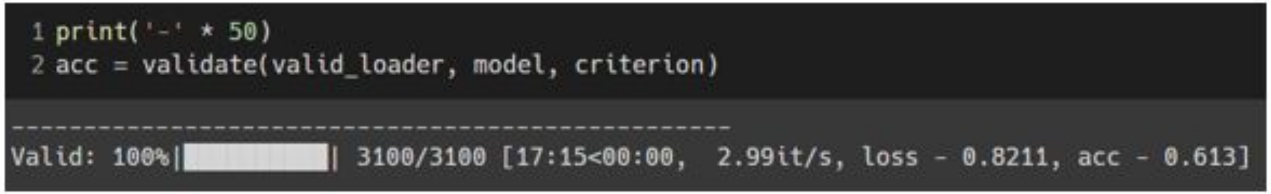
strong



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

```
--------------------------------------------------
Valid: 100%|██████████| 3100/3100 [33:15<00:00,  1.55it/s, loss - 1.9859, acc - 0.677]
```

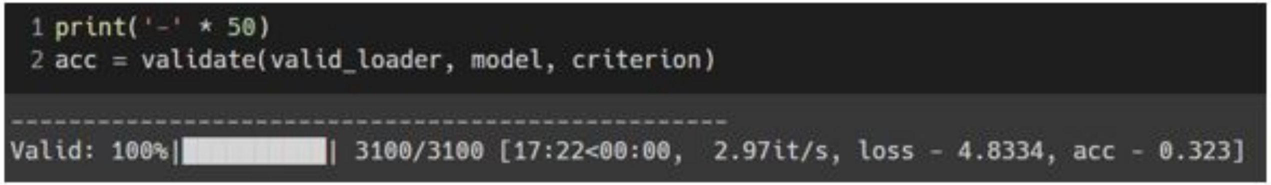
medium



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

```
--------------------------------------------------
Valid: 100%|██████████| 3100/3100 [17:15<00:00,  2.99it/s, loss - 0.8211, acc - 0.613]
```

weak

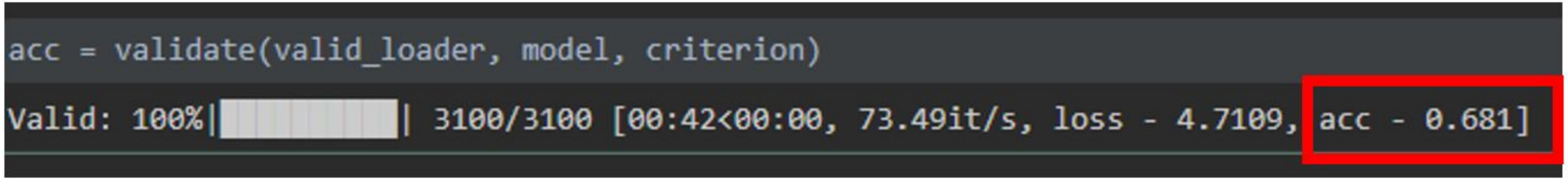


```
print('-' * 50)
acc = validate(valid_loader, model, criterion)
```

```
--------------------------------------------------
Valid: 100%|██████████| 3100/3100 [17:22<00:00,  2.97it/s, loss - 4.8334, acc - 0.323]
```

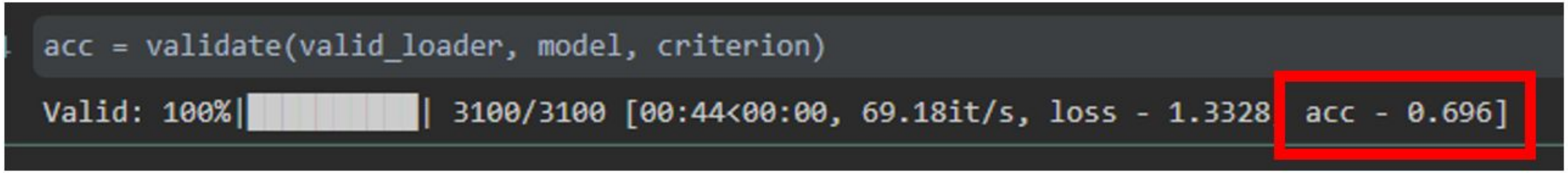
Sharpening noise test

강



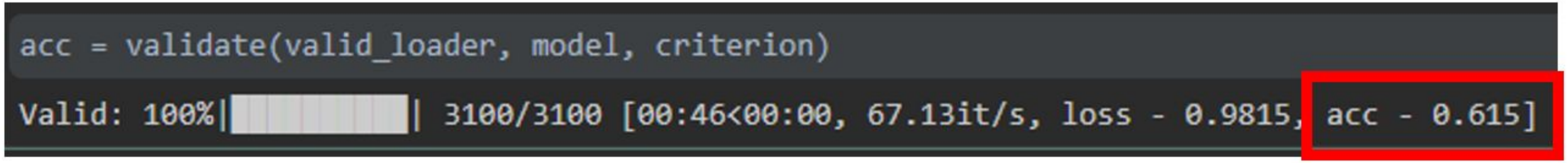
```
acc = validate(valid_loader, model, criterion)
Valid: 100%|██████████| 3100/3100 [00:42<00:00, 73.49it/s, loss - 4.7109, acc - 0.681]
```

중



```
acc = validate(valid_loader, model, criterion)
Valid: 100%|██████████| 3100/3100 [00:44<00:00, 69.18it/s, loss - 1.3328 acc - 0.696]
```

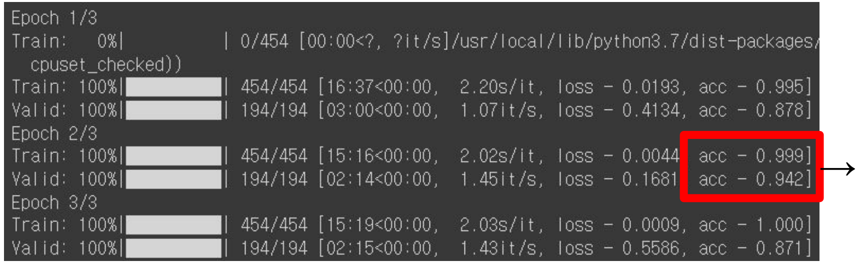
약



```
acc = validate(valid_loader, model, criterion)
Valid: 100%|██████████| 3100/3100 [00:46<00:00, 67.13it/s, loss - 0.9815, acc - 0.615]
```

Gaussian model adversarial train

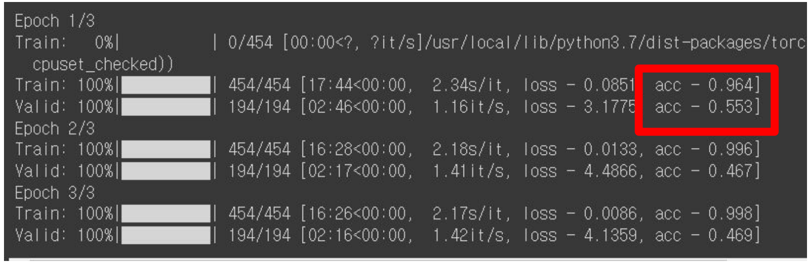
weak model



```
Epoch 1/3
Train:   0%|          | 0/454 [00:00<?, ?it/s]/usr/local/lib/python3.7/dist-packages/
  cpuset_checked))
Train: 100%|██████████| 454/454 [16:37<00:00,  2.20s/it, loss - 0.0193, acc - 0.995]
Valid: 100%|██████████| 194/194 [03:00<00:00,  1.07it/s, loss - 0.4134, acc - 0.878]
Epoch 2/3
Train: 100%|██████████| 454/454 [15:16<00:00,  2.02s/it, loss - 0.0044  acc - 0.999]
Valid: 100%|██████████| 194/194 [02:14<00:00,  1.45it/s, loss - 0.1681  acc - 0.942]
Epoch 3/3
Train: 100%|██████████| 454/454 [15:19<00:00,  2.03s/it, loss - 0.0009, acc - 1.000]
Valid: 100%|██████████| 194/194 [02:15<00:00,  1.43it/s, loss - 0.5586, acc - 0.871]
```

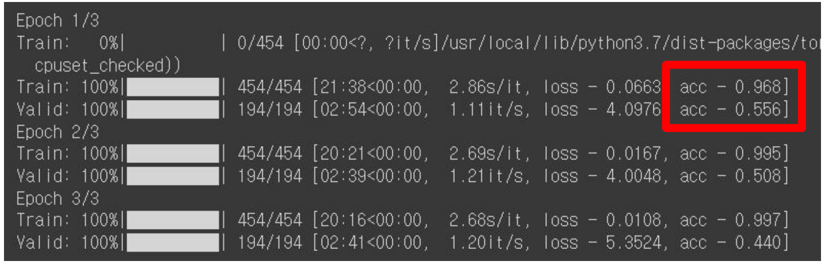
→ 최고 성능

medium model



```
Epoch 1/3
Train:   0%|          | 0/454 [00:00<?, ?it/s]/usr/local/lib/python3.7/dist-packages/torc
  cpuset_checked))
Train: 100%|██████████| 454/454 [17:44<00:00,  2.34s/it, loss - 0.0851  acc - 0.964]
Valid: 100%|██████████| 194/194 [02:46<00:00,  1.16it/s, loss - 3.1775  acc - 0.553]
Epoch 2/3
Train: 100%|██████████| 454/454 [16:28<00:00,  2.18s/it, loss - 0.0133, acc - 0.996]
Valid: 100%|██████████| 194/194 [02:17<00:00,  1.41it/s, loss - 4.4866, acc - 0.467]
Epoch 3/3
Train: 100%|██████████| 454/454 [16:26<00:00,  2.17s/it, loss - 0.0086, acc - 0.998]
Valid: 100%|██████████| 194/194 [02:16<00:00,  1.42it/s, loss - 4.1359, acc - 0.469]
```

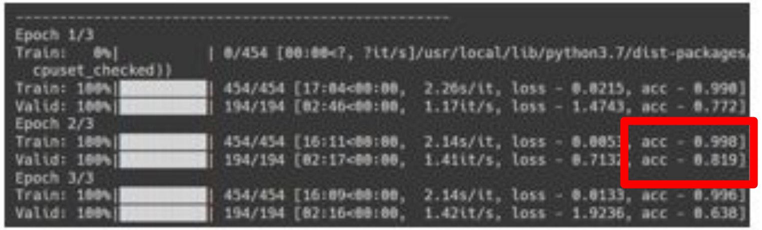
strong model



```
Epoch 1/3
Train:   0%|          | 0/454 [00:00<?, ?it/s]/usr/local/lib/python3.7/dist-packages/tor
  cpuset_checked))
Train: 100%|██████████| 454/454 [21:38<00:00,  2.86s/it, loss - 0.0663  acc - 0.968]
Valid: 100%|██████████| 194/194 [02:54<00:00,  1.11it/s, loss - 4.0976  acc - 0.556]
Epoch 2/3
Train: 100%|██████████| 454/454 [20:21<00:00,  2.69s/it, loss - 0.0167, acc - 0.995]
Valid: 100%|██████████| 194/194 [02:39<00:00,  1.21it/s, loss - 4.0048, acc - 0.508]
Epoch 3/3
Train: 100%|██████████| 454/454 [20:16<00:00,  2.68s/it, loss - 0.0108, acc - 0.997]
Valid: 100%|██████████| 194/194 [02:41<00:00,  1.20it/s, loss - 5.3524, acc - 0.440]
```

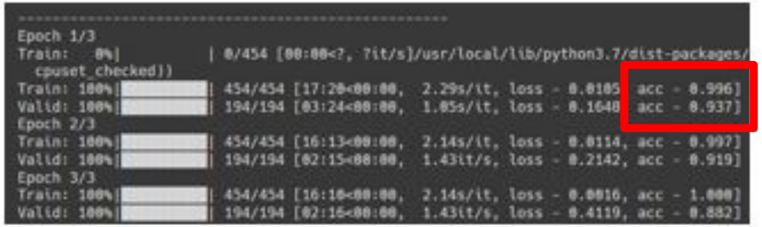
Salt and pepper model adversarial train

strong



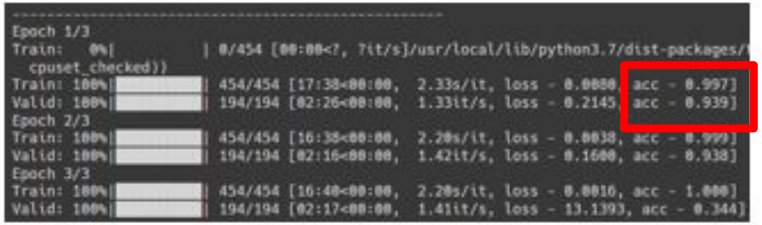
```
-----------------------------------------------------------------
Epoch 1/3
Train:   0%|          | 0/454 [00:00<?, ?it/s]/usr/local/lib/python3.7/dist-packages/
  cpuset_checked))
Train: 100%|██████████| 454/454 [17:04<00:00,  2.26s/it, loss - 0.0215, acc - 0.990]
Valid: 100%|██████████| 194/194 [02:46<00:00,  1.17it/s, loss - 1.4743, acc - 0.772]
Epoch 2/3
Train: 100%|██████████| 454/454 [16:11<00:00,  2.14s/it, loss - 0.0053, acc - 0.998]
Valid: 100%|██████████| 194/194 [02:17<00:00,  1.41it/s, loss - 0.7132, acc - 0.819]
Epoch 3/3
Train: 100%|██████████| 454/454 [16:09<00:00,  2.14s/it, loss - 0.0133, acc - 0.996]
Valid: 100%|██████████| 194/194 [02:16<00:00,  1.42it/s, loss - 1.9236, acc - 0.638]
```

medium



```
-----------------------------------------------------------------
Epoch 1/3
Train:   0%|          | 0/454 [00:00<?, ?it/s]/usr/local/lib/python3.7/dist-packages/
  cpuset_checked))
Train: 100%|██████████| 454/454 [17:20<00:00,  2.29s/it, loss - 0.0185 acc - 0.996]
Valid: 100%|██████████| 194/194 [03:24<00:00,  1.05s/it, loss - 0.1648 acc - 0.937]
Epoch 2/3
Train: 100%|██████████| 454/454 [16:13<00:00,  2.14s/it, loss - 0.0114, acc - 0.997]
Valid: 100%|██████████| 194/194 [02:15<00:00,  1.43it/s, loss - 0.2142, acc - 0.919]
Epoch 3/3
Train: 100%|██████████| 454/454 [16:10<00:00,  2.14s/it, loss - 0.0016, acc - 1.000]
Valid: 100%|██████████| 194/194 [02:16<00:00,  1.43it/s, loss - 0.4119, acc - 0.882]
```

weak

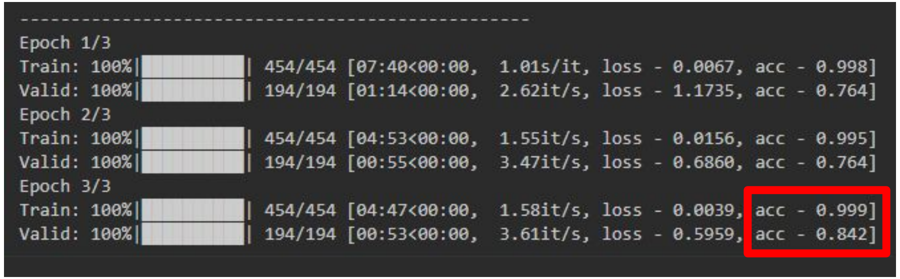


```
-----------------------------------------------------------------
Epoch 1/3
Train:   0%|          | 0/454 [00:00<?, ?it/s]/usr/local/lib/python3.7/dist-packages/
  cpuset_checked))
Train: 100%|██████████| 454/454 [17:38<00:00,  2.33s/it, loss - 0.0080, acc - 0.997]
Valid: 100%|██████████| 194/194 [02:26<00:00,  1.33it/s, loss - 0.2145, acc - 0.939]
Epoch 2/3
Train: 100%|██████████| 454/454 [16:38<00:00,  2.20s/it, loss - 0.0038, acc - 0.999]
Valid: 100%|██████████| 194/194 [02:16<00:00,  1.42it/s, loss - 0.1600, acc - 0.938]
Epoch 3/3
Train: 100%|██████████| 454/454 [16:40<00:00,  2.20s/it, loss - 0.0016, acc - 1.000]
Valid: 100%|██████████| 194/194 [02:17<00:00,  1.41it/s, loss - 13.1393, acc - 0.344]
```

→ 최고 성능

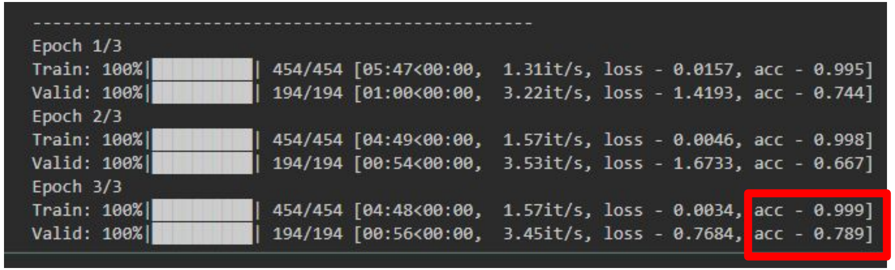
Sharpening model adversarial train

강



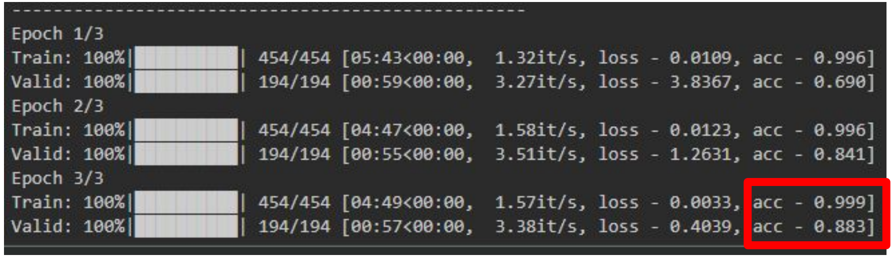
```
--------------------------------------------------
Epoch 1/3
Train: 100%|██████████| 454/454 [07:40<00:00,  1.01s/it, loss - 0.0067, acc - 0.998]
Valid: 100%|██████████| 194/194 [01:14<00:00,  2.62it/s, loss - 1.1735, acc - 0.764]
Epoch 2/3
Train: 100%|██████████| 454/454 [04:53<00:00,  1.55it/s, loss - 0.0156, acc - 0.995]
Valid: 100%|██████████| 194/194 [00:55<00:00,  3.47it/s, loss - 0.6860, acc - 0.764]
Epoch 3/3
Train: 100%|██████████| 454/454 [04:47<00:00,  1.58it/s, loss - 0.0039, acc - 0.999]
Valid: 100%|██████████| 194/194 [00:53<00:00,  3.61it/s, loss - 0.5959, acc - 0.842]
```

중



```
--------------------------------------------------
Epoch 1/3
Train: 100%|██████████| 454/454 [05:47<00:00,  1.31it/s, loss - 0.0157, acc - 0.995]
Valid: 100%|██████████| 194/194 [01:00<00:00,  3.22it/s, loss - 1.4193, acc - 0.744]
Epoch 2/3
Train: 100%|██████████| 454/454 [04:49<00:00,  1.57it/s, loss - 0.0046, acc - 0.998]
Valid: 100%|██████████| 194/194 [00:54<00:00,  3.53it/s, loss - 1.6733, acc - 0.667]
Epoch 3/3
Train: 100%|██████████| 454/454 [04:48<00:00,  1.57it/s, loss - 0.0034, acc - 0.999]
Valid: 100%|██████████| 194/194 [00:56<00:00,  3.45it/s, loss - 0.7684, acc - 0.789]
```

약



```
--------------------------------------------------
Epoch 1/3
Train: 100%|██████████| 454/454 [05:43<00:00,  1.32it/s, loss - 0.0109, acc - 0.996]
Valid: 100%|██████████| 194/194 [00:59<00:00,  3.27it/s, loss - 3.8367, acc - 0.690]
Epoch 2/3
Train: 100%|██████████| 454/454 [04:47<00:00,  1.58it/s, loss - 0.0123, acc - 0.996]
Valid: 100%|██████████| 194/194 [00:55<00:00,  3.51it/s, loss - 1.2631, acc - 0.841]
Epoch 3/3
Train: 100%|██████████| 454/454 [04:49<00:00,  1.57it/s, loss - 0.0033, acc - 0.999]
Valid: 100%|██████████| 194/194 [00:57<00:00,  3.38it/s, loss - 0.4039, acc - 0.883]
```

→ 최고 성능

	sharpening (strong)	**sharpening (medium)**	**sharpening (weak)**	**salt & pepper noise (strong)**	**salt & pepper noise (medium)**	**salt & pepper noise (weak)**
gaussian noise (strong)	loss – 19.4163 **acc – 0.323**	loss – 12.6805 **acc - 0.369**	loss - 9.0989 **acc - 0.443**	loss – 9.0557 **acc - 0.355**	loss – 6.1364 **acc - 0.435**	loss – 2.6379 **acc - 0.645**
gaussian noise (medium)	loss - 15.5655 **acc - 0.355**	loss - 10.0029 **acc - 0.455**	loss - 7.3472 **acc - 0.550**	loss – 3.8904 **acc - 0.481**	loss – 3.2610 **acc - 0.527**	loss – 1.9622 **acc - 0.689**
gaussian noise (weak)	loss – 4.7037 **acc - 0.677**	loss – 2.1664 **acc - 0.677**	loss – 1.2107 **acc - 0.786**	loss – 5.4284 **acc - 0.677**	loss – 3.0913 **acc - 0.677**	loss – 0.9826 **acc - 0.621**

	sharpening (strong)	**sharpening (medium)**	**sharpening (weak)**	**gaussian noise (strong)**	**gaussian noise (medium)**	**gaussian noise (weak)**
salt & pepper noise (strong)	loss - 10.4407 **acc - 0.362**	loss - 10.0229 **acc - 0.412**	loss - 9.1598 **acc - 0.452**	loss - 2.1381 **acc - 0.375**	loss - 4.0276 **acc - 0.4**	loss - 6.2126 **acc - 0.483**
salt & pepper noise (medium)	loss - 5.5641 **acc - 0.393**	loss - 6.3789 **acc - 0.425**	loss - 6.2602 **acc - 0.450**	loss - 0.7999 **acc - 0.665**	loss - 2.6169 **acc - 0.403**	loss - 7.1634 **acc - 0.372**
salt & pepper noise (weak)	loss - 61.9681 **acc - 0.323**	loss - 60.1062 **acc - 0.323**	loss - 52.1954 **acc - 0.323**	loss - 58.9012 **acc - 0.323**	loss - 59.3096 **acc - 0.323**	loss - 45.0288 **acc - 0.323**

	gaussian noise (strong)	gaussian noise (medium)	gaussian noise (weak)	salt & pepper noise (strong)	salt & pepper noise (medium)	salt & pepper noise (weak)
sharpening (strong)	loss – 2.08 **acc – 0.323**	loss –3.09 **acc - 0.323**	loss -1.99 **acc - 0.553**	loss –13.65 **acc -0.323**	loss –12.54 **acc - 0.323**	loss – 5.65 **acc - 0.352**
sharpening (medium)	loss - 0.67 **acc -0.733**	loss - 0.73 **acc -0.736**	loss - 1.68 **acc - 0.676**	loss – 6.07 **acc - 0.337**	loss – 5.44 **acc - 0.353**	loss – 2.02 **acc -0.639**
sharpening (weak)	loss – 2.51 **acc -0.388**	loss – 3.46 **acc - 0.383**	loss – 4.45 **acc - 0.389**	loss – 4.96 **acc - 0.41**	loss – 8.32 **acc - 0.323**	loss – 4.96 **acc -0.41**

**눈여겨볼 점: sharpening medium모델의 gaussian noise에 대한 성능이 눈에 띄게 높음

전반적인 결과

- test set: real noise이미지 1000장, fake noise 이미지 2100장
- **adversarial training한 모델** 중 모두 **weak모델 성능이 제일 높음**
- 대다수 모델의 성능이 strong, medium보다 **weak 데이터셋에 대해서 높음**
- real 이미지 거의 다 맞추고, **fake이미지에서 성능 대폭 하락**

  → noise real 데이터는 맞추고, noise fake이미지는 왜 못맞추는가?
- **전반적으로 낮은 성능**

  → 생성한 모델이 general하게 강인한 모델X

더 실험해보고 싶은 부분

- **두 가지 이상의 노이즈로 학습**을 시켜보기

  → 한가지로 학습시켰을 때보다는 더 general한 성능이 나올 수도 있겠다

- xception이 아닌 **다른 네트워크 사용**해보기 ()
- **테스트셋**에 아예 **다른 데이터** 사용해보기

  → 일반화되었을 가능성

참고 링크: https://www.mdpi.com/1999-5903/13/11/288/htm#B1-futureinternet-13-00288