

# Cogsys Winter Code Hack: 3회차

## End-to-end pipeline with collaboration



Daehyun Cho

Cognitive System Lab

A.I. Dept, Korea Univ.

## 전체적인 계획

1. 대회 참여하기 & GitHub Repository 개설하기D
2. Pytorch DataLoader 구축하기 & Code Review하기
3. Pytorch Model 개발하기 & 제출해보기
4. 여러 가지 실험을 하기 위해 세팅하기 (w/ Hydra) & wandb logging하기
5. Pytorch-Lightning으로 갈아타보기
6. Accuracy 100% 만들기

이 과정을 조금 불친절하게 진행할 예정

## 지향

- ✓ 알잘딱깔센
- ✓ Collaborations
- ✓ 조금이라도 모르면 구글링 (선무당이 사람잡음)

## 지양

- ✗ Copy & Paste: 단순한 syntax정도는 괜찮은데, 전체 파이프라인을 베껴오는 건 취지를 아주 많이 벗어남
- ✗ 나의 온 시간을 투자하기: 짬짬히... 틈틈히 진행하는 것을 권장 (교수님한테 혼나기 싫음)

## TODO List

- ✓ Create flake8 workflow in your github
- ✓ Find 'pytorch-image-models' library and see how to load pre-trained model
- ✓ Go to 'paperswithcode' and see who is solving image classification task
- ✓ Load pre-trained model and edit models with respect to your input/output
- ✓ Make a train code
- ✓ Make a validation/inference code
- ✓ Submit your submission

During todos...

- ✓ Commit middle works as much as you can (make it unconscious)
- ✓ Push your work to remote repository



- ☒ 지난 시간에 만든 Dataloader와 plotting code를 .py 파일에 잘 스크립트화했는지 확인해봅시다.
- 아래와 같을 필요 없고 같을 수 없습니다. 유사한 구조로 불러올 수 있는지 확인해보세요.

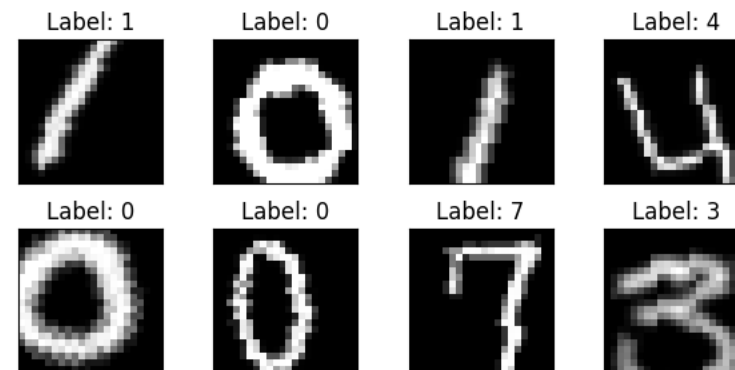
```
from torch.utils.data import DataLoader

from digitrec.plot_utils import plot_tensors
from digitrec.dataloader import DigitDataset

dataset = DigitDataset(data_dir="./data", file_name="train.csv")
dataloader = DataLoader(dataset, batch_size=16)

batch = next(iter(dataloader))

plot_tensors(X=batch[0].squeeze(), y=batch[1])
```





- 첫 시간에 만든 github workflows를 만들어봅시다 (Flake8 만들던거)
- GitHub Actions에서 에러가 나지 않았는지 확인해봅시다.
  - 에러가 났다면 어떻게 고칠지 생각해봅시다.
  - 대표적으로 빈 라인에 tab이 들어가면 안되는데 들어간 경우가 있습니다.
  - 에러가 나면 고쳐줍시다
  - 사실 잘 지키지 않아도 되는데 굳이 굳이 flake8에서 뭐라하는 에러들이 있습니다. 이런 걸 제외하려면 ``.flake8`` 이라는 flake8의 configuration 파일을 만들어서 관리해줄 수 있습니다.
    - 그래도 어지간하면 지키는게 코드 일관성+가독성에 도움이 됩니다.

```
16 ./digitrec/dataloader.py:11:1: W293 blank line contains whitespace
17 ./digitrec/dataloader.py:30:1: W293 blank line contains whitespace
18 ./digitrec/dataloader.py:39:32: W292 no newline at end of file
19 Error: Process completed with exit code 1.
```



timm

Pytorch-image-models

- 이제 단순한 classification의 경우 직접 모델을 더 개발할 필요는 없습니다.
- 이미 잘 나온 모델이 너무 많아요. 대표적으로 아래 라이브러리에 어지간한 모델 구조와 pre-trained weights가 저장되어 있습니다.
  - <https://github.com/rwightman/pytorch-image-models>
- 여기서 불러다가 내 데이터에 fine-tune해도 충분합니다.
- ☒ Timm 설치 후 ResNet를 불러옵시다
  - 🔍 How to load? <https://timm.fast.ai/>
  - 나중에 Swin이나 다른 ViT 계열의 transformer 모델들도 불러서 실험해보세요.
  - 모델의 parameters 수를 세봅시다. 얼마나되나요? <https://gist.github.com/1pha/577206868960cb4ca3146e737dd7a319>
- 우리의 28X28 이미지 입력을 잘 넣어봅시다.
  - ☒ 0-9의 digit을 맞춰야하기 때문에 출력은 10개의 값이 나와야 합니다.





## Train models

- ☒ 이제 주어진 모델을 학습해봅시다. 앞에서 만든 dataloader와 model을 통해서 만들어줄 수 있습니다.
- 전체적인 프로세스 [https://tutorials.pytorch.kr/beginner/basics/quickstart\\_tutorial.html](https://tutorials.pytorch.kr/beginner/basics/quickstart_tutorial.html)
  - Loop dataloader
    - Fetch a single batch
    - Make predictions through model
    - Calculate loss
    - Backward loss





- 뭔가 이상합니다. Test data를 바로 prediction해서 내보내자니, 이 모델이 일을 제대로 하고 있는건지 검증할 필요가 있을 것 같습니다.
  - 애초에 train data를 전부 다 가지고 훈련해봤으니 당연히 train data에서는 잘 나올건데, 한 번도 본 적 없는 test data에서도 잘 나올까요?
  - <https://blog.roboflow.com/train-test-split/>
- 이걸 확신할 수 없으니 train.csv에 있는 42,000건의 케이스를 일부 쪼개서 validation으로 사용해야 합니다.
  - 42,000건을 train/valid로 나눠서 train으로만 학습을 진행하고, valid로 중간검증을 합니다.
  - 실제로 여러 대회에서는 제출 횟수에 제한을 두기 때문에 이 train/valid로 쪼개주는 작업을 잘해주어야 합니다.
-  Scikit-learn에는 train\_test\_split이라는 함수가 있습니다. 이를 통해서 나눠보세요.
-  처음부터 DigitDataset에서 나눠주면 얼마나 좋을까요? Dataset 내부에서 train/valid를 나눠주는 코드를 구현해주세요.



## Wrap up


Put your code in scripts

- 열심히 .ipynb에 넣었던 코드를 .py 스크립트에 넣어서 정리해줍시다.
- 오늘 첫 장표에 나왔던 코드처럼 import 몇 줄 만에 훈련과 제출까지 나올 수 있도록 만들어줍시다.
- ☒ 함수화로 먼저 진행해봅시다
- ☒ 진행해보고 나면 관리하기 힘들게 전역변수가 너무 많은 것을 알 수 있습니다. Class에 넣어서 정리해봅시다.
- ☒ 학습이 끝나면 모델을 저장하는 코드까지 넣어봅시다. [https://tutorials.pytorch.kr/beginner/saving\\_loading\\_models.html](https://tutorials.pytorch.kr/beginner/saving_loading_models.html)
- 처음부터 하기엔 어려워서 코드를 참고하면서 진행하시면 좋을 것 같네요.



# Make Submission

## Test submission

-  Test data도 불러와서 최종결과물을 만들어줍니다.
  - Dataloader는 기본적으로 내부의 데이터를 섞어줍니다. 안 섞이게 만들어주려면 어떻게 해야 할까요?
  - <https://pytorch.org/docs/stable/data.html#torch.utils.data.DataLoader>
  - sample\_submission.csv에 맞춰서 결과물을 만들어주세요
  - 그리고 결과물을 kaggle에 제출해보세요.

All

Successful

Errors

Recent ▼

Submission and Description

Public Score ⓘ



**test\_submission.csv**

Complete · now

**0.96264**



- 학습 중간에 결과물을 확인하는 것은 너무 중요합니다.
  - 현재 학습이 정상적으로 되고 있는지
  - 원하는 데이터가 잘 나오고 있는지
  - 출력값은 의도대로 나오고 있는지 (0-9 digit classification이면 10개의 확률값을 잘 뱉는지)
- 여러 설정값들을 쑤셔넣는 것도 다른 실험을 할 때마다 조절하기 힘듭니다.
  - Batch\_Size, model, optimizer ...,
- 이 온갖 것들은 하나하나는 사소한 것처럼 보이지만 전부 다 넣어준다고 생각하면 아주 귀찮은 일들입니다.
- 다음 시간에 **hydra**와 **wandb**를 이용해서 진행해볼 예정입니다.

Thank you 🙏

Daehyun Cho

1phantasmas@korea.ac.kr