# 부스트캠프 랩업 리포트 가이드

## 캠퍼 랩업 리포트 가이드 및 예제

랩업 리포트는 대회 기간 동안 내가 시도한 기술적인 도전, 학습과정에서의 교훈, 마주한 한계와 도전숙제 등을 담아 정리하면서, 학습의 주체로서 무엇을 해봤고, 어떻게 해봤고, 무엇을 얻었는지에 대해 스스로 회고하고 하나의 논리적인 흐름의 콘텐츠를 만드는 데 목적이 있습니다. 실무에서는 일을 잘하는 것 못지 않게 논리적으로 커뮤니케이션하고, 지속적인 기록을 기반으로 성장을 이어가는 것이 보편적입니다. AI 기술학습에 대한 학습을 통해 실무에서 일을 잘하는 여러분을 기대하며, 아래의 가이드를 참고하여 리포트를 작성해보세요.

### **<**기술적인 도전**>**

#### 본인의 점수 및 순위

LB 점수 0.6561, 56등 (반드시 포함해 주세요)

- 검증(Validation) 전략
  1. 제공된 val data json 파일을 이용하였고, val mIoU 측정법은 batch_size마다 mIoU를 구하는 baseline code에서 토론게시판을 참고하여 모든 이미지에 대한 mIoU를 구하는걸로 바꿈.
  2. k-fold할 땐 멘토님이 나눠주신 json파일을 이용해서 val mIoU를 계산함

- 사용한 모델 아키텍처 및 하이퍼 파라미터
  1. 아키텍처: SegNet vgg16
     a. LB 점수 : 0.4126
     b. training time augmentation
        i. Normalize
     c. 한 이유
        i. SegNet이 처리속도와 경량화에 중점을 둔 모델이라고 해서 개인적인 관심으로 사용함.
        ii. 실습 과제로 만든걸 활용해보고 싶었음.
        iii. vgg16이 그래도 몇번 만져본 모델이라 친숙해서
     d. 추가시도
        i. Normalize 안했을 때 LB 0.4093이라 Normalize 하는게 이득이라는걸 알아냄.
        ii. pretrained 된걸 불러오는게 훨씬 좋더라. val mIoU가 0.2160에서 0.3467로 상승했었음.

2. 아키텍처: Unet mobilenetv2
    a. LB 점수 : 0.4647
    b. training time augmentation
        i. Normalize
    c. optimizer : Adam
    d. 한 이유
        i. 전 프로젝트에서 무거운 모델로만 실험했었는데 훈련시킬때마다 시간이 너무 많이 들어 여러 실험을 못했었다. 그래서 이번엔 비교적 가볍고 성능도 적당한 걸 찾아서 여러 실험을 하려고 함.
    e. 추가시도
        i. eval loss가 제일 낮게 나온거랑 eval mIoU가 제일 높게 나온거랑 어떤게 더 좋나 비교해봄. eval loss가 제일 낮게 나온 7 epoch 에선 LB 0.4647이 나왔지만 eval mIoU가 제일 높게 나온 19 epoch에선 LB 0.4462가 나옴. 이 때 eval mIoU를 계산하는 코드가 문제임을 알고 수정.. 했지만 애초에 LB에서 mIoU를 구하는 방식이 이상한 거였음.
        ii. opencv에서 물체 경계면을 그리는 Canny를 섞어서 사용하면 어떨까 했는데 성능이 감소함.

3. 아키텍처: DeepLab V3+ se-resnext101_32xd
    a. LB 점수 : 0.6309
    b. training time augmentation
        i. CropNonEmptyMaskIfExists(256)
        ii. CLAHE
        iii. HorizontalFlip
        iv. Rotate
        v. Normalize
    c. optimizer : Adam
    d. 한 이유
        i. 같은 팀원이 DeepLab V3+ se-resnext가 잘 나왔다고 해서 제일 무거운걸로 시험해봄.
        ii. 팀원의 여러 augmentation 실험 결과 제일 잘 나온걸로
    e. 추가시도
        i. 토론게시판을 보고 Rotate가 있으면 잘 나온다고 해서 사용함. 또 이 Rotate를 어느 위치에 두느냐에 따라서도 성능이 미묘하게 차이가 났었는데, 맨 처음부분에 Rotate를 하는게 성능이 아주 조금 더 잘나왔었음. 아마 mask crop과 연관되어 있는것 같다.
        ii. TTA를 HorizentalFlip, Scale 2개로 해봤는데 안했을때의 원래 LB는 0.6238에서 max를 사용했을 땐 LB 0.6014로, sum을 사용했을 땐 LB 0.6044로 감소. 좋을 줄 알았는데 의외로 역효과였다.

4. 아키텍처: DeepLab V3+ timm-regnety_120
    a. LB 점수 : 0.6118
    b. training time augmentation
        i. CropNonEmptyMaskIfExists(256)

        ii. CLAHE
        iii. HorizontalFlip
        iv. Normalize
    c. optimizer : Adam
    d. 한 이유
        i. DeepLab V3+ 가 제일 좋은 모델이라고 판단, backbone을 바꿔서 어떤 backbone이 더 좋을지 팀원끼리 나눠서 실험했는데, 난 파라미터수가 많으면서 내 GPU인 P40이 감당할 만큼의 것을 고름
    e. 추가시도
        i. epoch 20의 PB가 eval loss가 가장 낮을때였을거다. epoch를 40까지는 돌려봐야 된다는 말을 듣고 더 돌려봤으나 성능이 떨어졌음.
5. 아키텍처: Unet++ inceptionv4
    a. LB 점수 : 0.5356
    b. training time augmentation
        i. CropNonEmptyMaskIfExists(256)
        ii. CLAHE
        iii. HorizontalFlip
        iv. Normalize
    c. optimizer : Adam
    d. 한 이유
        i. backbone인 inceptionv4에서 학습 이미지에 background도 들어있고 우리 훈련 데이터에도 background 구분하는게 있으니까 도움이 되지 않을까 실험
    e. 추가시도
        i. epoch 20의 PB가 eval loss가 가장 낮을때였을거다. epoch를 40까지는 돌려봐야 된다는 말을 듣고 더 돌려봤으나 성능이 떨어졌음.

6. 아키텍처: DeepLab V3+ mobilenet v2
    a. LB 점수 : 0.568
    b. training time augmentation
        i. CropNonEmptyMaskIfExists(256)
        ii. CLAHE
        iii. HorizontalFlip
        iv. HueSaturationValue
        v. Normalize
    c. optimizer : Adam
    d. 한 이유
        i. backbone인 inceptionv4에서 학습 이미지에 background도 들어있고 우리 훈련 데이터에도 background 구분하는게 있으니까 도움이 되지 않을까 실험
    e. 추가시도
        i. epoch 20의 PB가 eval loss가 가장 낮을때였을거다. epoch를 40까지는 돌려봐야 된다는 말을 듣고 더 돌려봤으나 성능이 떨어졌음.

ii. 색깔 바꾸는 color jitter도 예전에 써본적이 있어서 같은 기능을 하는 HueSaturationValue를 사용해봄. 근데 eval mIoU는 낮게 나와서 사용 안했는데 PB는 더 높게 나왔음.

- 앙상블 방법
  1. soft label 덧셈 방법을 사용.

- 시도했으나 잘 되지 않았던 것들
  1. pseudo labels 방법을 시도해 봤으나 hard pseudo label을 사용했을 땐 val mIoU가 0.11정도가 될 정도로 성능이 너무 처참하게 떨어져서 soft pseudo label을 했는데 epoch가 진행될수록 LB이 0.01씩 낮아졌음. 내가 잘못했거나 테스트 데이터가 이상하거나 논문을 잘못 선택했거나 한 이유인것 같다. 아마 논문을 잘 못 선택한듯.
  2. TTA를 해봤지만 성능이 내려간 것 같다. TTA 결과 mask를 띄워서 확인하고 넣은건데도 그렇다. 이미지 1~2장 봤다고 판단하지 말고 val mIoU를 확인하는게 중요한 듯. 근데 시간이 없어서 못햇으니까..

## <학습과정에서의 교훈>

### 학습과 관련하여 개인과 동료로서 얻은 교훈 (구체적인 상황을 기반으로)

### 피어세션을 진행하며 좋았던 부분과 동료로부터 배운 부분

논문을 볼 때 paperswithcode 사이트에서 내 목적에 맞는 논문을 찾아보자. 내가 처음에 찾아서 작성해본 meta pseudo label은 보편적인 classification에 pretrained model도 원래 정확도가 높아 사용 가능한 방식이 아니였을까 생각함. 나중에 보니 내가 여기서 한 image segmentation용 pseudo label 논문이 있더라.

ipynb로 되어있는 baseline을 IDLE로 바꾸는데 시간이 생각보다 오래걸린다. 생각해놓고 있자.

역시 sota모델을 활용하는게 제일 좋은것 같다. 결과도 보고 여러 kaggle과 dacon에서 우승한 모델은 뭘 썼나 보며 배우자.

다음 대회때는 무조건 모델 개조해서 사용해보기. 지금까지 보면 이 모델을 직접적으로 개조하고 추가해서 사용한게 성능이 매우 좋았었음.

여러 좋은 사이트들을 배웠음.

models library

qubvel/segmentation_models.pytorch

Augmentation

Streamlit

논문

Papers with Code - Semantic Segmentation

백그라운드 실행

리눅스 백그라운드 파이썬 실행 nohup 사용법

tmux 입문자 시리즈 요약

전까지는 개인전이어서 경쟁하는 느낌이 강했는데, 이번 팀전도 팀끼리만 싸우는 느낌이 들었지만 후반부에 다른 팀들도 모여 정보를 주고받는걸 보고 경쟁보단 모두 모여 지성집단으로 하나의 문제를 해결한다는 느낌이 들었다.

할 때마다 엄청난 사람들은 항상 내가 생각하는 것 이상의 생각을 해낸다. 예를들면 난 단순히 pseudo label 할려고 test data를 그냥 넣었지만, 멘토님은 test data에 대한 모델 결과를 보고 어떤걸 학습에 넣으면 좋을지 일일이 보며 추가하는 모습, 모델 개조해서 뒤에 attention을 넣는 등. 항상 배운다.. 근데 나도 해야한다..

## <마주한 한계와 도전숙제>

### 아쉬웠던 점들

1. pseudo label 실패.
2. 모델 개조 상상만 해보고 못해봄.

### 한계/교훈을 바탕으로 다음 스테이지에서 새롭게 시도해볼 것

1. pseudo label 성공시켜보기.
2. 모델 개조 해보기