

# Wrap up Report

---

본 Wrap up Report는 부스트캠프 P stage 1 Image classification task를 수행하면서 있었던 모든 내용을 담고 있습니다.

## 기술적인 도전

---

### 본인의 점수 및 순위

- LB 점수 acc : 73.9841% / f1score : 0.6738 / 152위로 최종 마무리!

## 검증 전략

1. Single model : 기본적으로 제공된 train 데이터셋을 0.2의 비율로 검증셋으로 활용하였습니다.
2. 5-fold cross validation : 위와 같은 방식이지만 동일 seed에서 서로 독립된 5개의 validation set을 설정하여 교차검증을 시행했습니다.

## 모델

- 최종 제출 모델 :
  - 아키텍처 : EfficientNet-b0
  - 전처리 및 autgmentation
    - centercrop (384,384)
    - img\_size : (512,384)
  - optimizer : Adam
  - scheduler : stepLR - gamma=0.794
    - 시작 Learning rate 0.001 , 20에폭 마지막 Learning rate 0.00001
  - Loss : cross-entropy

## 시도했던 방법들

- 아키텍처 :
  - EfficientNet-b0~b6 :

baseline의 backbone로는 b0모델을 주로 사용하였습니다. 동일 조건하에 모델의 사이즈가 커질수록 같은 수의 에폭에서 validation 데이터 기준 더 좋은 성능을 발휘하는 것 같았습니다. 하지만, 대회 말미에는 거의 b0를 주로 사용하였는데, 분포가 너무 imbalance하여 검증셋을 제대로 구성하지 않으면 제대로 '검증'이 되지 않음을 느꼈기 때문입니다.(좋은 validation acc를 보유한 모델이 제출하였을 경우 더 낮거나, 나쁜 모델이 더 좋은 public score를 기록하는 경우가 많았습니다.)
  - ResNext50 :

토론 글에서 본 Timm의 ClassifierHead를 이용하여 마스크, 성별, 나이 각각에 대한 브랜치를 구성하고 따로 학습 후 inference에서 합치는 방식을 위해 사용해봤던 모델입니다. 변인을 통제하지 못한 채 독립적으로 사용했던 모델이었기에 아키텍처 자체의 효과는 잘 모르겠습니다.

- Loss :

- focalloss :

분류 에러에 근거하여 loss에 가중치를 주는 방식으로 학습되는 loss이기에 어느정도 imbalance한 데이터에 효과적일 것으로 생각했지만, 그닥 효과를 보지 못했습니다.

모든 조건을 동일하게 했을 때, Cross-entropy의 경우 0.67의 f1스코어였던 반면, focal loss를 사용했을 경우 0.59로 많이 하락했던 결과가 있었습니다.

당시에는 이 두번이 비교를 통해 당연히 cross-entropy가 더 나에게 맞다고 생각했지만, 데이터 샘플링을 효과적으로 하지 못했던 것이 더 좋은 성능을 낼 수 있었음에도 그렇지 못한 원인이 되었던 것 같습니다.

- Data handling:

- age control(증강):

남녀 모두에 대하여, 나이 클래스 3개 중 60대 이상에 대한 데이터가 거의 8배가량 차이나는 것을 확인할 수 있었습니다.

이를 완화하기 위해 60대 남녀 모두에 대하여 각각 데이터를 증강시켜 학습에 적용했었습니다. 다른 하이퍼파라미터 및 모델을 타이트하게 학습시키지 않았음에도 나쁘지 않은 성능을 보여주었습니다.

하지만, '가장' 좋은 모델과의 성능차이가 있었기에 사용하지 않았습니다.

- age label modify(레이블 필터링):

위와 같이 분포에 대한 이슈를 인지하고 있었기에 이를 보완하고자 나이의 경계를 수정하여 적용했습니다. [60대 이상]에 대한 경계를 [58세 이상]으로 레이블링하였고, 유의미한 성능 변화를 얻을 수 있었습니다.

## 시도하려 했으나 못했던 방법들

- Coral-cnn :

single모델에 브랜치만을 3개로 나누는 것이 아니라, 세 개의 모델을 각각 만들어 사용하지 못한 것이 아쉽습니다. age 모델은 특히 따로 pretrained 된 모델과 아키텍처를 사용해보려 했으나 시간 부족으로 시도하진 못했습니다.

- data sampler :

불균형한 데이터에서 검증셋의 정의가 매우 중요하다는 것을 뒤늦게 깨달았기에 시도하지 못했습니다.

## 성능 개선 효과를 보지 못했던 방법들

- augmentation:

아무런 augmentation을 하지 않은 모델이 가장 좋은 성능을 냈습니다. 기존 512,384 에서 resize(224)로 했던 모델에서 시작하여 점차적으로 input image 사이즈를 키웠을 때 성능이 좋았지만, Randomhorizontalflip 등 augmentation 방식을 추가할 수록 성능이 하락하였습니다.

- kFold cross validation:

앞서 언급했듯, 교차검증의 효과를 보지 못하였습니다. 아마도 좋은 성능을 냈던 모델은 public dataset에 맞는 분포를 학습할 수 있는 train / valid dataset split이 운 좋게 되었다고 생각합니다.

## 대회를 진행하면서 깨달은 점

---

대회에 참가하면서 개인적으로, 그리고 팀의 한 팀원으로서 얻은 교훈이 많습니다.

- 우선 개인적으로 대회를 진행하면서 성취한 것들이 많았습니다.
- 첫째, 파이토치에 대한 이해입니다.

저는 이전까지 텐서플로우와 케라스를 위주로 머신러닝 역량을 쌓아왔습니다. 개발 역량이 부족하다고 막연히 생각했었지만, 딱히 파이토치에 대한 필요성을 크게 느끼지 못했습니다. 그 이유는 별다른 어려움 없이 원하는 모델을 설계할 수 있었기 때문입니다. 하지만, 파이토치를 조금 더 이론적인 부분이 아닌 실전에 적용하는 경험을 통해 python의 클래스에 대한 이해와, Tensor의 관점에서 모델의 구조를 바라볼 수 있는 능력을 기를 수 있는 원동력이 된 것 같아 매우 만족스럽습니다.

- 둘째, 기록의 중요성입니다.

대회를 진행하면서 많은 양의 모델을 통한 추론 결과를 제출하는 과정속에서 혼란스러운 제 자신을 발견할 수 있었습니다. 기록을 안하진 않았지만, 매우 추상적이고 그때 기입하는 그 당시의 감정과 간략한 내용만을 담은 기록을 작성했었습니다. 하지만, 2주라는 짧은 시간임에도 많은 시도와 다양한 결과를 구분하는 것은 매우 어려웠고, 변인통제가 되지 않은 막연한 실험결과는 모델을 개선하는데 아무런 도움이 되지 않았습니다. 급할수록 돌아가라는 말의 의미를 깨달은 2주였습니다.

- 반면에 팀원과 함께하는 캠퍼들 중 한명의 동료로서 얻은 교훈은 '진정한 의미의 협업'이 무엇인지를 조금이나마 깨닫는 계기가 되었다는 점입니다.

피어세션을 진행하면서 90분이라는 짧은 시간동안 서로의 아이디어와 시행착오들을 공유하는 것은 대회 기간 중 매우 의미있는 시간이었습니다. 특히, 처음에 매일 바뀌는 피어세션 멤버가 적응되지 않아 다소 불편했습니다. 서로의 시도나 도전하는 것들에 대한 피드백을 다시 받지 못하기 때문입니다. 하지만, 일주일이나 지나면서 생각이 바뀌었습니다. 지나온 일주일간의 매번 바뀐 25명의 캠퍼분들께 얻은 정보들과 의견들, 그리고 피드백은 제 점수를 향상시키는데 많은 도움이 되었기 때문입니다.

가령, 무차별적으로 augmentation 방법을 추가하고 있던 제게 어느 한 캠퍼분께서 말씀해주신 "augmentation을 아예 하지 않거나 centercrop정도만 하라"고 했던 조언은 직접적으로 성능 향상에 도움이 되었습니다.

반대로 피어세션에서 만난 몇 분들께 제 시행착오와 피드백들을 공유했을 때, 피어세션을 진행하면서 에러 디버깅을 하신분들도 있었고, '그건 생각지도 못했다'고 했던 분도 계셨습니다.

이처럼 서로 부족하지만, 많은 사람들이 모여 서로의 의견을 나누는 것은 매우 중요하다는 것을 깨달았던 경험이었습니다.

## 마주한 한계와 도전과제들

---

### 아쉬운점

- 베이스라인 구현을 했지만, 효과적으로, 그리고 빠르게 구현하지 못했던 점이 아쉽습니다.
- 변인 통제를 하지 않은 상태로 마구잡이식의 실험들이 성능향상에 도움되지 않는다는 것을 너무 늦게 깨달았습니다.
- 부족한 개발 역량으로 인해 많은 error를 만났고, 이를 고치는데 많은 시간이 걸렸습니다. 이제는 해결할 수 있는 문제들이라고 생각하지만, 온전히 대회에 집중할 시간이 줄어들었던 점은 매우 아쉽습니다.
- 검증 방법에 대한 내용을 심각하게 고민하지 못했던 점이 성능 향상에 많은 걸림돌이 되었습니다. 분포에 맞는 검증 데이터를 구성하는 것의 중요성을 막연하게 인지하고 있었지만, 이를 해결하지

못했고 결국 아무리 높은 검증 데이터 정확도가 나와도 믿지 못하는 사태에 도달하는 원인이 되었습니다.

## 도전 숙제

- 위의 아쉬웠던 점들을 개선하는 방향으로 진행할 예정입니다.
- 조금은 귀찮고, 오래걸리더라도 하나하나 꼼꼼히 기록하며 변인조건을 컨트롤하면서 성능향상에 힘쓸 것입니다.
- 토론헤시판에서 많은 도움을 받았습니다. 이후 스테이지에서는 다른 캠퍼분들께 먼저 공유하고 함께 도울 수 있도록 노력할 예정입니다.