

## ○ 프로젝트 개요

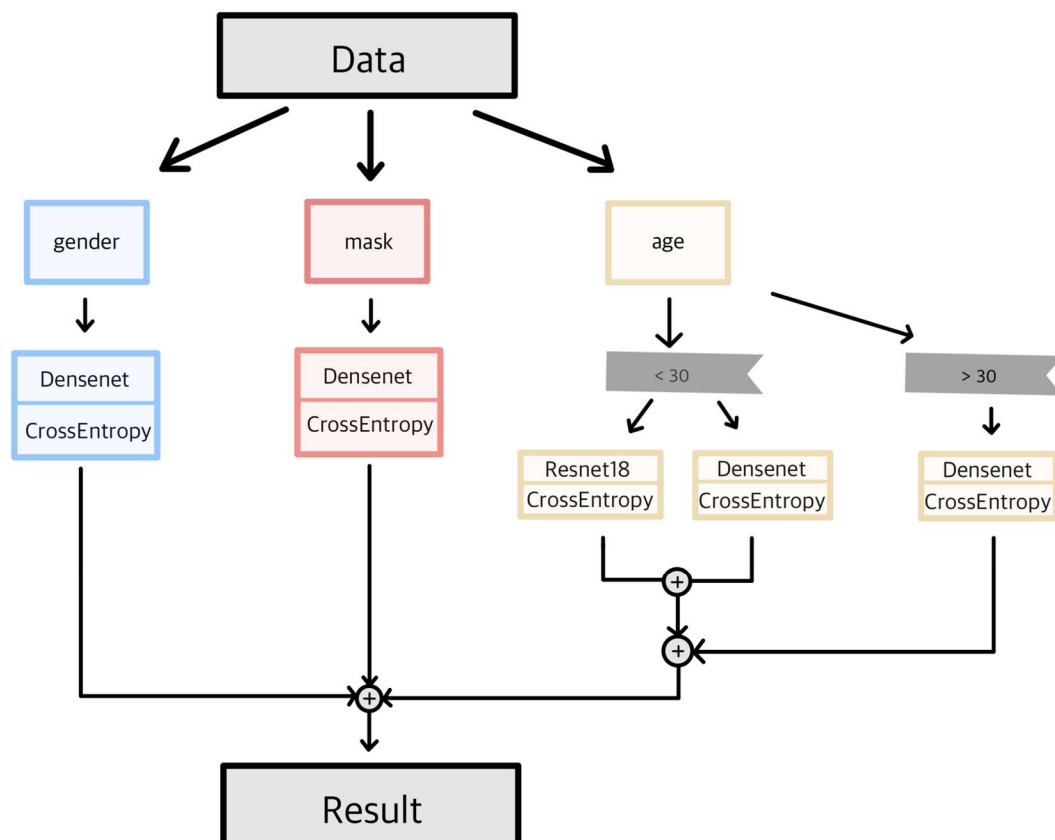
**주제** : 마스크의 착용 여부 및 나이와 성별 분류

**개요** : 각각 task별로 모델링을 하여 최종 결과 label 값에 맞춰서 예측

- Git - 코드 및 결과 값 공유
- Gpu - v100
- python - 3.8.5 , torch - 1.10.2

**기대 효과** : 코로나로 인한 건물 입출입 시 사람의 특징 및 마스크 착용 여부를 판단하여 감염을 예방할 수 있다.

**프로젝트 구조** :

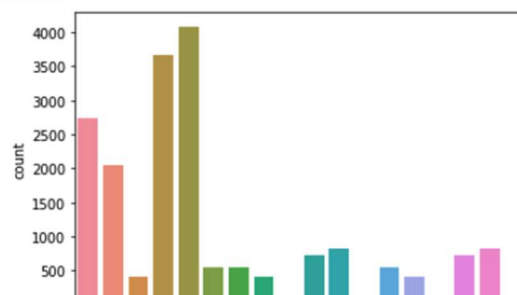


## 데이터 셋 개요

Task	Class	Label
Age	10 ~ 29	0
	30 ~ 59	1
	60 ~	2
Mask	Wear	0
	Incorrect	1
	Not wear	2
Gender	Male	0
	Female	1
Final Label	$6 \times \text{mask} + 3 \times \text{gender} + \text{age}$	0~17

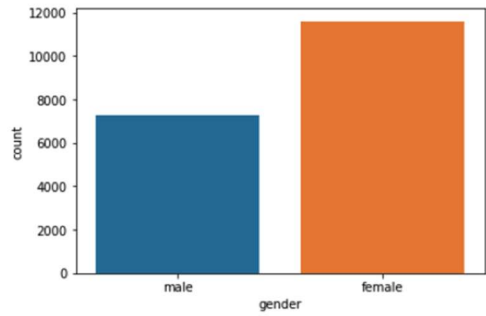
### All classes

```
sns.countplot(data=data_label_df, x='label')
plt.show()
```



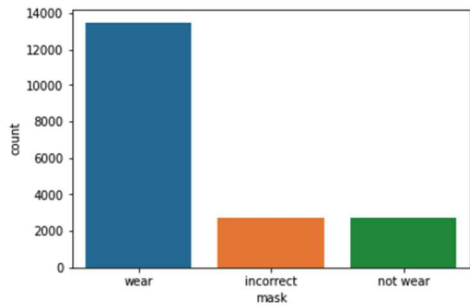
Gender

```
sns.countplot(data=converted_df, x='gender')
plt.xticks([0,1],['male','female'])
plt.show()
```



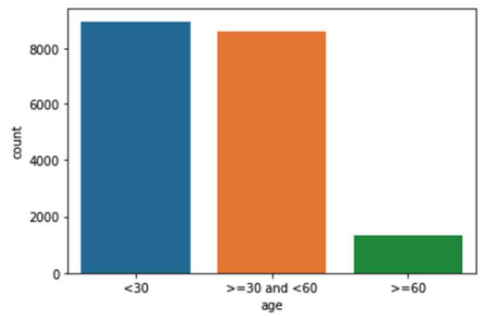
Mask

```
sns.countplot(data=converted_df, x='mask')
plt.xticks([0,1,2],['wear','incorrect','not wear'])
plt.show()
```



Age ¶

```
sns.countplot(data=converted_df, x='age')
plt.xticks([0,1,2],['<30','>=30 and <60','>=60'])
plt.show()
```



## ○ 프로젝트 팀 구성 및 역할

대회 첫날에는 모델링과 data augmentation을 세명, 두명이 각각 맡아서 하기로 했다.

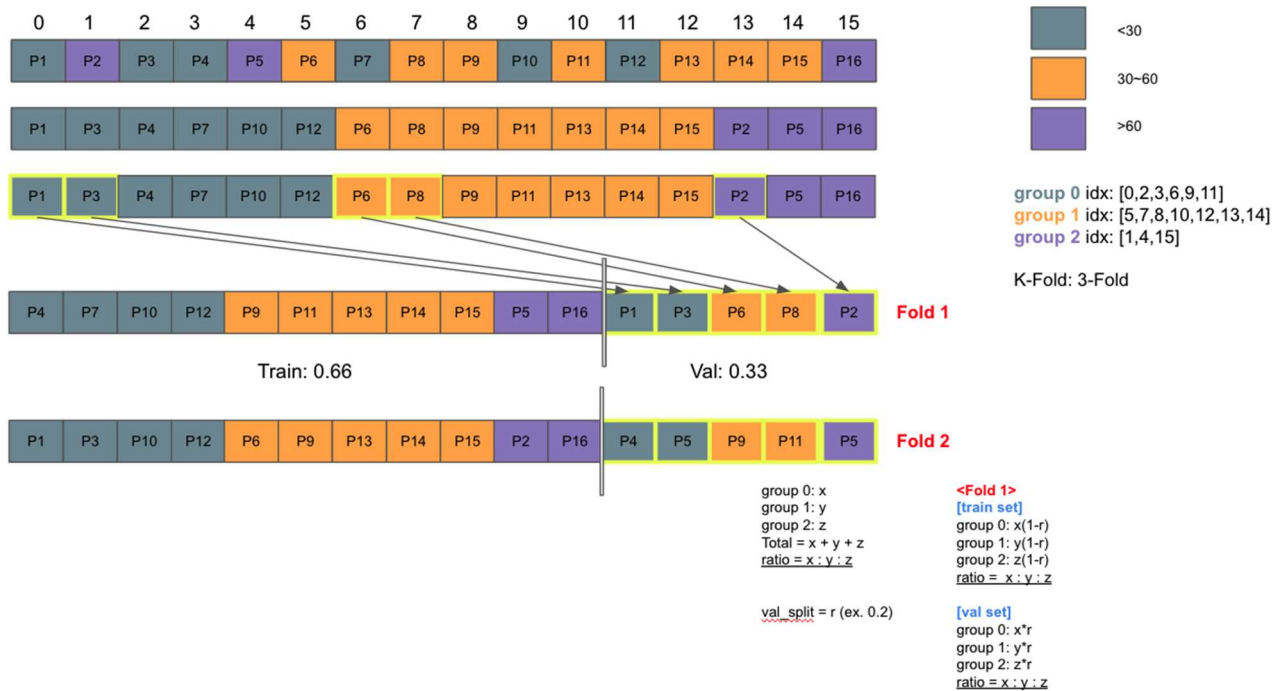
하지만 data augmentation은 오랜 시간동안 할 수 있는게 많이 없었고 각자 모델링을 하고 싶은 욕심이 커서 후반부에는 각자 모델링을 하면서 떠오르는 여러가지 아이디어들을 미팅시간에 공유했다. Age, gender, mask를 통합해서 18개 클래스로 분류하는 모델, age, gender, mask를 각각 분류하는 모델, gender와 mask만 분류하는 모델, 그리고 age만

분류하는 모델 등 다양한 방법을 시도해 봤었고 이 중 성능이 잘 나온 모델들을 github에 공유하면서 ensemble 적용시에 취합해서 제출 했었다.

데이터 전처리로 배경을 제거하는 semantic segmentation과 face crop을 해줬고 관련 코드들을 깃헙에 올려 다른 팀원들이 간편하게 사용할수 있었다.

## ○ 프로젝트 수행 절차 및 방법

처음에 EDA를 진행해서 데이터의 분포를 파악하고, mask, age, gender를 나눠서 예측하자는 의견이 나왔다. 18개 클래스를 한번에 예측하는 것보다 각각의 task에 대해서 학습하는것이 좀 더 수월할 것이라는 생각이었다. 각 task에 대해서 여러가지 모델들을 학습시켜 앙상블 해줬고 추합하여 최종 prediction을 해주었다. 하지만 validation performance와 Public LB performance 갭이 너무 컸다. 후에 임의로 train/validation set을 나누게 되면 한사람의 이미지들이 train/val set에 중복해서 들어간다는 문제점을 깨달았고 이를 해결하기 위해서 각 사람과 나이를 기준으로 training set과 validation set을 나눠주었다. 기존에 제공된 데이터셋 폴더가 person을 기준으로 되어있는데 age class가 3개가 있으므로 각 age group에 person 폴더를 매칭 시켜주고 각 age group에 K-Fold를 적용해서 train/val set을 나눠주었다. 이를 통해 train/val set의 mask, age, gender분포를 일정하게 유지시켜 줄 수 있었고 이후 조금 더 객관적인 validation performance를 측정 할 수 있었다.



이유 다양한 아이디어를 공유하며 시도해보았고, 내용은 아래와 같다.

- 1) torchvision의 여러 모델중 pre-train 했을 때, 높은 성능을 얻는 모델 찾기
  - ResNet, EfficientNet, DenseNet, RegNext
- 2) 모델 Ensemble 해서 최적의 결과 찾기
- 3) Loss 다양하게 적용하기
  - Binary Cross Entropy, Cross Entropy, Focal
- 4) 데이터 셋을 나눠서 학습하기
  - MTCNN을 이용하여 얼굴 crop하기, fcn-resnet50 이용한 segmentation 이후 face crop하기, segmentation face crop 이후 upper face만 crop 하기.



(original image)

(face crop image)

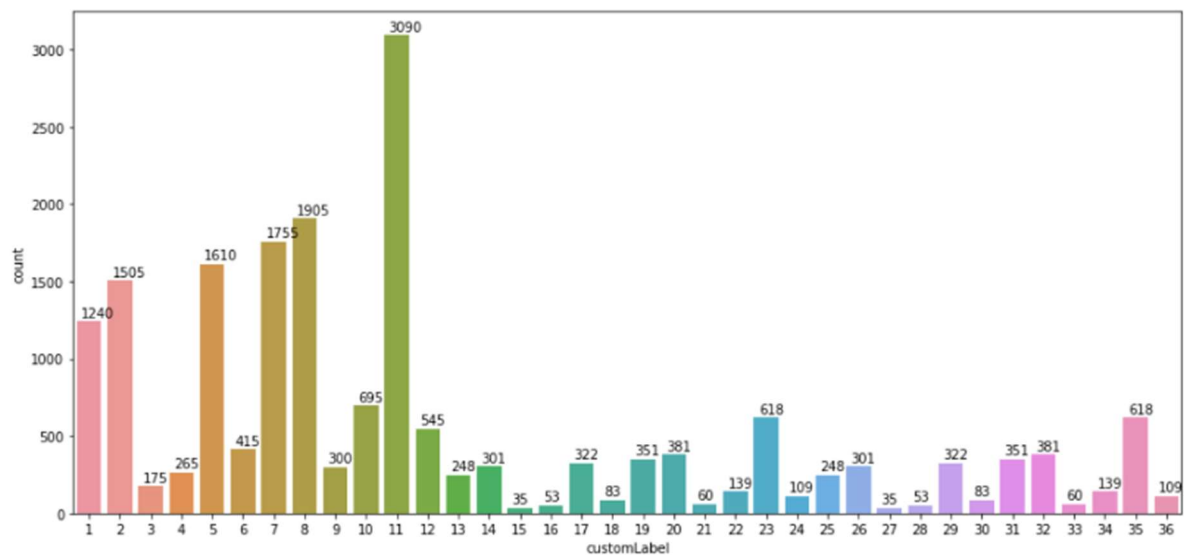
(segmentation image)

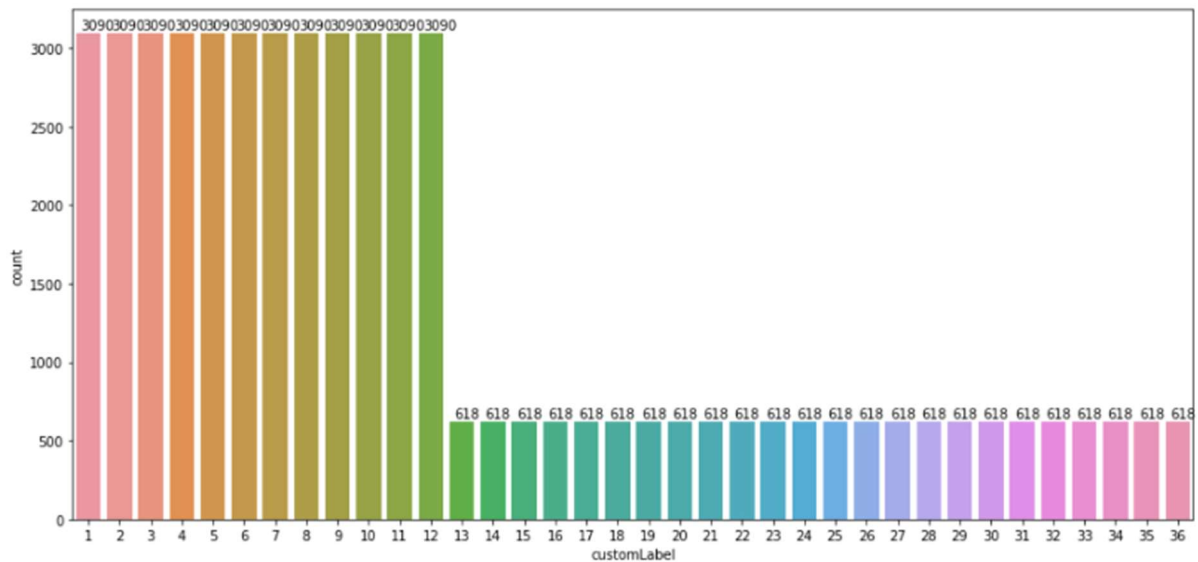
(upper face crop)

- 5) mask, gender, age 순으로 예측하며 이전 예측 결과를 활용하여 상응하는 모델에서 결과값 얻기
- 6) age group을 3개가 아닌 6개로 나눠서 학습하기
- 7) 데이터 imbalance를 해결하기 위해 mix하기



- age를 label을 3개 아닌 6개로 labeling해서 imbalance 해결.





LB Score	f1 score	accuracy	rank
public	0.7525	79.4921	19th
private	0.7463	78.6984	15th

## ○ 자체 평가 의견

### - 좋았던 점

모두 대회는 처음 경험하여 익숙하지 않은 상태에서 진행하다보니 처음에는 조금 헤맸지만 팀원 서로서로 알려주고 배우면서 서로의 실력 향상에 도움을 주었다.

상대적으로 어려운 age label 문제 해결을 위해 model ensemble, softmax 확률값 조정, custom loss function , cutmix 등등 다양한 아이디어로 접근해가며 성능을 높여가고 좀 더 일반화된 결과를 도출할 수 있었다.

## - 아쉬운 점

아쉬웠던 점은 대회 초반에 탄탄한 협업 세팅을 하지 못해 대회 기간 동안 협업을 많이 하지 못했었다. 팀원들 각자 제출을 위한 처음부터 끝까지의 과정을 겪고 결과물을 내고 싶은 마음이 있어 개인이 진행하고 제출하게 되었다. 하지만 대회 기간이 다소 지나고나서 부터는 각자 개인이 진행했던 실험 결과물들과 아이디어들을 서로 공유하면서 피드백을 주는 과정에서 많은것을 배울수 있었다.

## - 더 알아갈 것

imbalance data의 upsampling과 undersampling을 하는 경우 훈련 data를 늘리는 upsampling이 효과적일 것이라고 생각했지만 이번 대회에서 age label은 undersampling으로 학습하였을 때 결과가 좋았다.

60대 미만과 이상의 label의 경우 구분자체가 어려운 경우가 있고 이 때 더 많은 data들이 overfitting이 일어나게된다고 생각된다.

이러한 data의 경우 imbalance를 맞춰주는 것보다 label간의 관계가 명확한 것이 더 좋지 않을까 생각된다.