

# Mask Image Classification - 19조 BCAA

## (P-Stage01) Mask Image Classification

- 1. 프로젝트 개요
  - 1-1.프로젝트 주제
  - 1-2. 프로젝트 개요
  - 1-3. 활용 장비 및 재료
  - 1-4. 기대 효과
- 2. 프로젝트 팀 구성 및 역할
- 3. 프로젝트 수행 절차 및 방법
- 4. 프로젝트 수행 결과
  - 4-1. 탐색적 분석 및 전처리
  - 4-2. 모델 개요
  - 4-3. 모델 선정 및 분석
  - 4-4. 모델 평가 및 개선
  - 4-5. 시연 결과 (Public 0.714점 / Private 0.705점)
  - Success
  - Fail
- 5. 자체 평가 의견
  - 5-1. 잘한 점들
  - 5-2. 시도했으나 잘 되지 않았던 것들
  - 5-3. 아쉬웠던 점들

## (P-Stage01) Mask Image Classification

### 1. 프로젝트 개요

#### 1-1.프로젝트 주제

- 효과적인 마스크 착용 상태 / 성별 / 나이 판별에 대한 시스템 구현 ⇒ 다양한 Image Classification Techniques를 활용

#### 1-2. 프로젝트 개요

- 구현내용
  - 이미지 분류 모델을 활용한 마스크 착용 / 성별 / 나이에 대해 총 18개의 클래스로 나눠 판별하는 전체 시스템 구현
- 컨셉
  - AI Stages 플랫폼 기반으로 하여 Competition 형태로 시스템 성능과 LB Score 간의 간극을 줄여나가는 프로세스
- 교육과의 관련성
  - 선수과목으로 수강한 딥러닝, PyTorch 강의들을 종합하여 실제 프로젝트에 적용, 전체적인 파이프라인을 모듈화해보는 과정 경험

#### 1-3. 활용 장비 및 재료

- [OS] Linux version 4.4.0-59-generic
- [CPU / GPU] Intel(R) Xeon(R) Gold 5220 CPU @ 2.20GHz / Tesla V100-SXM2-32GB
- [협업툴] Git / Slack / Notion
- [IDE] VSCode / Pycharm / Jupyter lab

#### 1-4. 기대 효과

- General
  - low-level 로 직접 코드를 구현해보면서 customizing 을 통한 딥러닝 구현 체득 과정
- Project

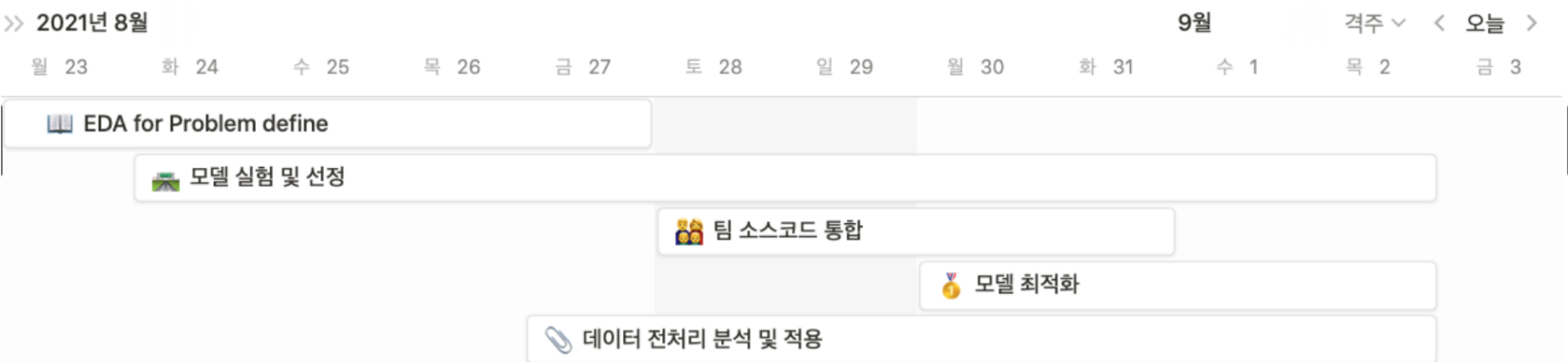
- 궁극적으로 정확한 마스크 착용 / 성별 / 나이 판별을 통한 코로나 확산 예방 효과

## 2. 프로젝트 팀 구성 및 역할

### 프로젝트 팀 구성 및 역할

Aa BCAA	≡ 역할
오재석_T2133 (팀리더)	- 팀원들이 제공한 아이디어에 다양한 hyper parameter 및 토론게시판에 나온 아이디어들 실험
최현진_T2234	- 모델 실험을 위한 코드 완성 - 성능 향상 방법 적용 및 제시 Regression K-Fold multi-model
박진영_T2096	- baseline 코드 기반 구조 제공 - 이미지 분류 관련 다양한 techniques 적용 및 공유
이종혁_T2171	- Hyper Parameter Tunning 실험을 통한 최적값 도출
김진용_T2063	- face crop, cutmix 클래스별 분류 등 데이터 전처리에 대한 분석을 통해 성능 향상에 기여
공은찬_T2009	- cosine loss 와 cross entropy를 combination 하여 새로운 loss function 제시
윤영훈_T2142	- train data label 비율 조절을 통한 성능 향상 (e.g) undersampling

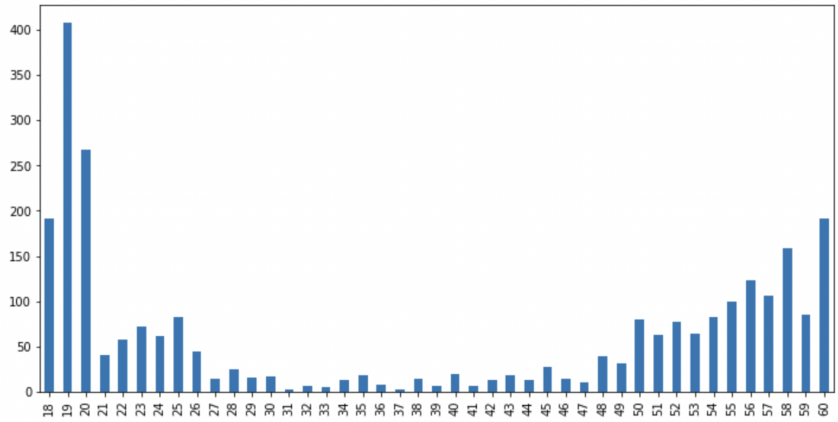
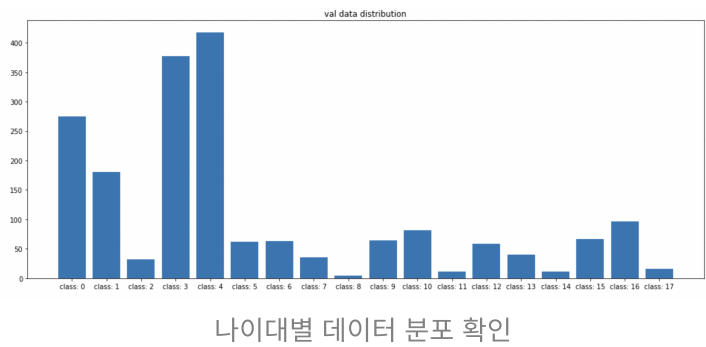
## 3. 프로젝트 수행 절차 및 방법



## 4. 프로젝트 수행 결과

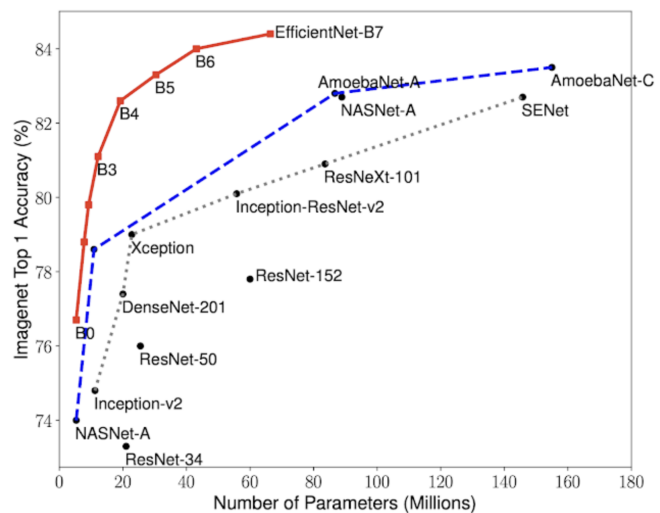
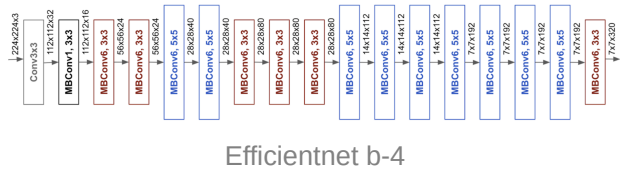
### 4-1. 탐색적 분석 및 전처리

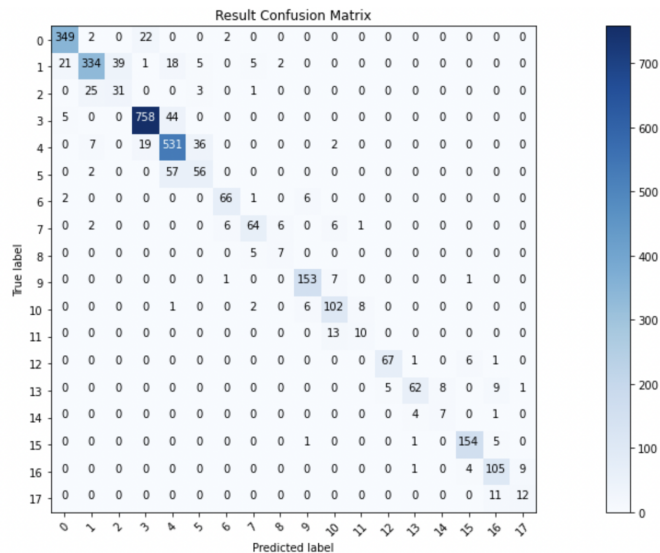
피쳐 데이터 컬럼별 Value Count 확인 => 불균형 확인



- Train
  - General : K-Fold를 사용하지 않고 학습하는 경우 데이터를 8:2 비율로 나누어 loss 및 accuracy로 확인
  - K-Fold : K-Fold 기법을 적용한 경우 데이터셋 전체 학습, Stratified K-Fold를 사용해 클래스 비율을 맞춰 validation set 선정
- Validation
  - Accuracy, Loss, F1 score, Recall, Confusion Matrix, Classification Report

### 4-2. 모델 개요





[결과 도식화] Task divide - mask / gender / age

## Fail

- Regression
  - 나이 모델을 회귀 방법으로 접근해 봤지만 학습된 모델의 결과를 매핑해주는 과정에서 경계값 조절이 필요한데 평가 데이터에 대한 경계값을 알 수 없음
- Model Ensemble
  - custom model에 2개 이상의 pre-trained 모델을 2개를 넣어 Fully Connected layer를 생성해 한번에 2가지 모델을 학습시켜 봤지만 성능변화 크게 없음
- Oversampling
  - 부족한 60세 이상 데이터를 증가시키고 transform 함수를 적용하여 클래스 불균형 문제를 해결하고자 했지만 성능 변화는 크지 않았음

## 5. 자체 평가 의견

### 5-1. 잘한 점들

- 팀원들과 전처리, Model, Hyperparameter, Optimizer 등 다양한 Task 를 분배하여 실험하고 공유한점
- 앞으로 진행될 프로젝트에서 사용할 수 있는 Template 소스코드를 완성할 수 있었던점

### 5-2. 시도했으나 잘 되지 않았던 것들

- 클래스 불균형 문제를 해결하기 위한 데이터 Oversampling 및 Undersampling 기법을 적용하였으나 성능 향상은 되지 않음
- Confusion Matrix 분석 결과 모델이 나이를 제대로 예측하지 못하는 경우가 많아 나이 예측 모델만 여러개 생성하여 앙상블 적용하였지만 성능 향상은 되지 않음
- 나이를 분류하는 문제를 Regression으로 접근하였고 자체 test set은 성능이 향상 되었지만 evaluation 데이터에는 성능이 향상되지 않았음
- 외부 데이터 활용 ⇒ 대회 종료일 시도하였으나 종료 시간과 맞물려 제출 하지 못함

### 5-3. 아쉬웠던 점들

- 기술
  - Seed를 유지하지 않고 여러개의 모델을 학습시키고 Ensemble을 시도해보지 못한점
  - 기존의 모델에서 문제점을 찾기보다 계속해서 새로운 시도에 오랜시간 머물러 있었던 점
  - 이미 모듈화가 되어있기 때문에 Flask Framework 에 API 로 추가해볼 수 있었으나 시간 관계상 실현해보지 못한 점
- 일반
  - 성능을 올리는데 집중하다보니 협업툴을 이용해서 통일된 소스나 데이터를 빠르게 적용하지 못했던 부분이 아쉬움
  - 코드 공유를 할 때 다른 팀원이 쉽게 사용할 수 있도록 재현성을 높이지 못한 것이 아쉬움