

# **Masked face detection Using CNN & Machine learning**

통계학과 212STG33 황예진

# 목차

**01**

## **Introduction**

- 주제 소개
- 선행 연구

**02**

## **Modeling**

- 전체 모델 구조
- 모델 구성 요소

**03**

## **Experiments**

- Using DenseNet
- Using MobileNetV2
- Using EfficientNetV2

**04**

## **Conclusion**

- 결론 및 한계점

# 1. Introduction

주제 소개

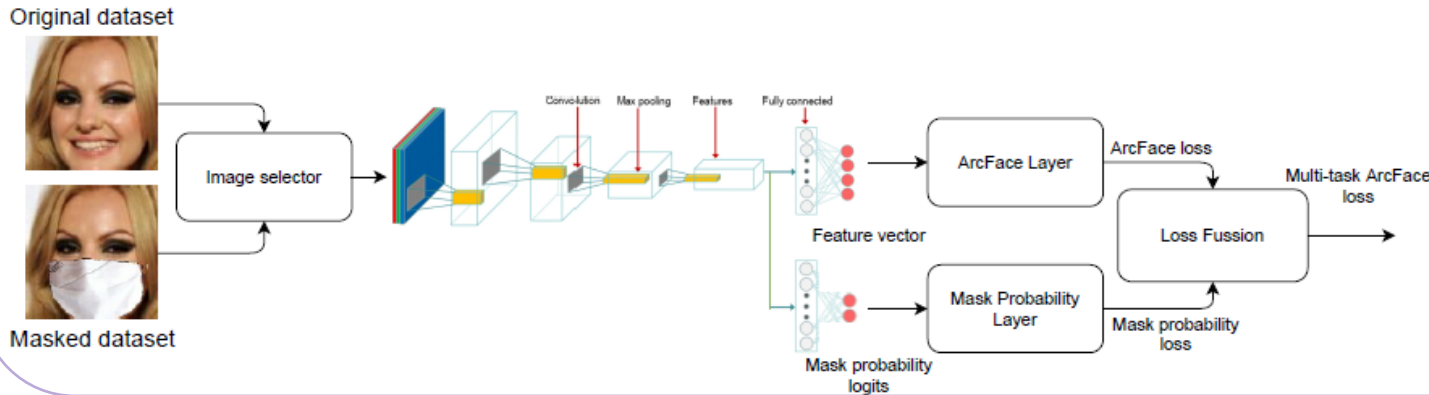
연구주제

## Masked face detection Using CNN & Machine learning

- 2020년 코로나19 바이러스의 발생 이후 유행의 장기화로 인한 마스크 착용의 일상화
- 실외 마스크 필수 착용이 해제된 이후에도 실내에서의 마스크 착용은 필수적
- 실내 시설을 이용하는 사람들이 마스크를 착용하고 있는지에 대해 판별
- 사람의 얼굴 데이터가 주어졌을 때 그 사람이 마스크를 쓰고 있는지 아닌지 예측

# 1. Introduction 선행 연구

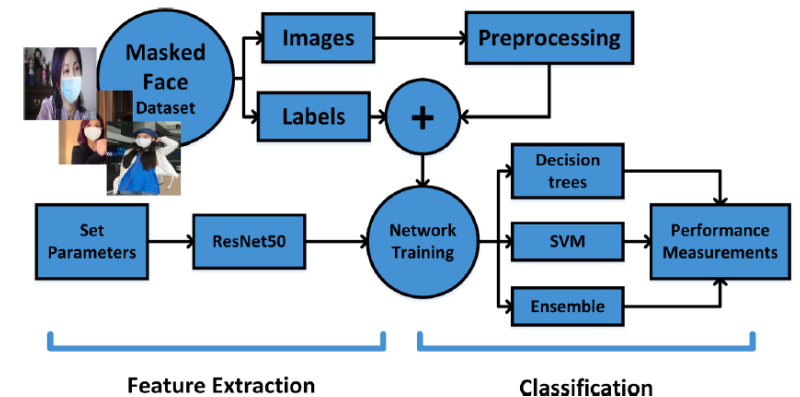
## - Boosting Masked Face Recognition with Multi-Task ArcFace (2021)



- ArcFace 방법론을 바탕으로 한 Multi-Task ArcFace(MTArcFace) 제안
- ResNet50을 backbone으로 사용
- ArcFace loss + mask probability logit을 바탕으로 구한 mask probability loss

## - A hybrid deep transfer learning model with machine learning methods For face mask detection in the era of the COVID-19 pandemic(2021)

- 딥러닝과 머신러닝 방법론의 하이브리드 모델을 제안
- Feature extraction: cnn 모델을 이용하여 이미지의 feature 추출
- Classification: Decision tree, SVM, ensemble 모델을 통해 각각 마스크 착용 여부를 예측하고 결과를 voting하는 stacking ensemble 사용



# 1. Introduction 선행 연구

- Boosting Masked Face Recognition with Multi-Task ArcFace (2021)
- A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic(2021)



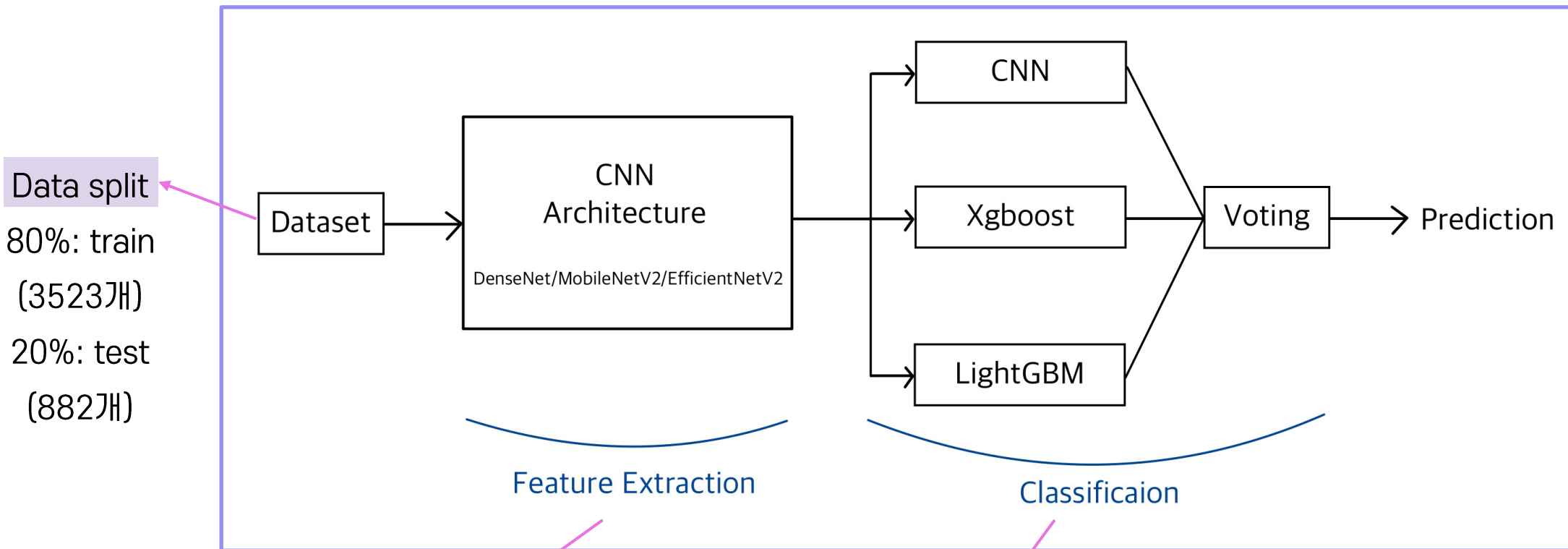
- Feature extraction을 위해 CNN 모델 사용
- 분류를 위해 CNN이 아닌 다른 구조를 가진 모델을 함께 사용



최근의 CNN 모델을 통해 feature extraction + 머신러닝 방법론 중 boosting을 이용하여 Classification

## 2. Modeling 전체 모델 구조

CNN과 boosting 방법을 이용한 stacking ensemble 모델 제안



### Feature extraction

DenseNet, MobileNetV2, EfficientNetV2를 각각 사용해보고 성능 비교

### Classification

Feature extraction에서 사용한 CNN 모델, XGBoost, LightGBM으로 각각 예측 후 stacking

## 2. Modeling 모델 구성 요소

### DenseNet

각 layer의 feature map 이 모든 다른 레이어의 feature map 과 연결되어 있는 구조.

### MobileNetV2

Depthwise separable convolution의 구조를 이용하여 연산량을 크게 감소시킨 CNN 모델 구조

### EfficientNetV2

모델의 width, depth, resolution을 최적의 비율을 조합하는 compound scaling을 사용하는 모델 구조

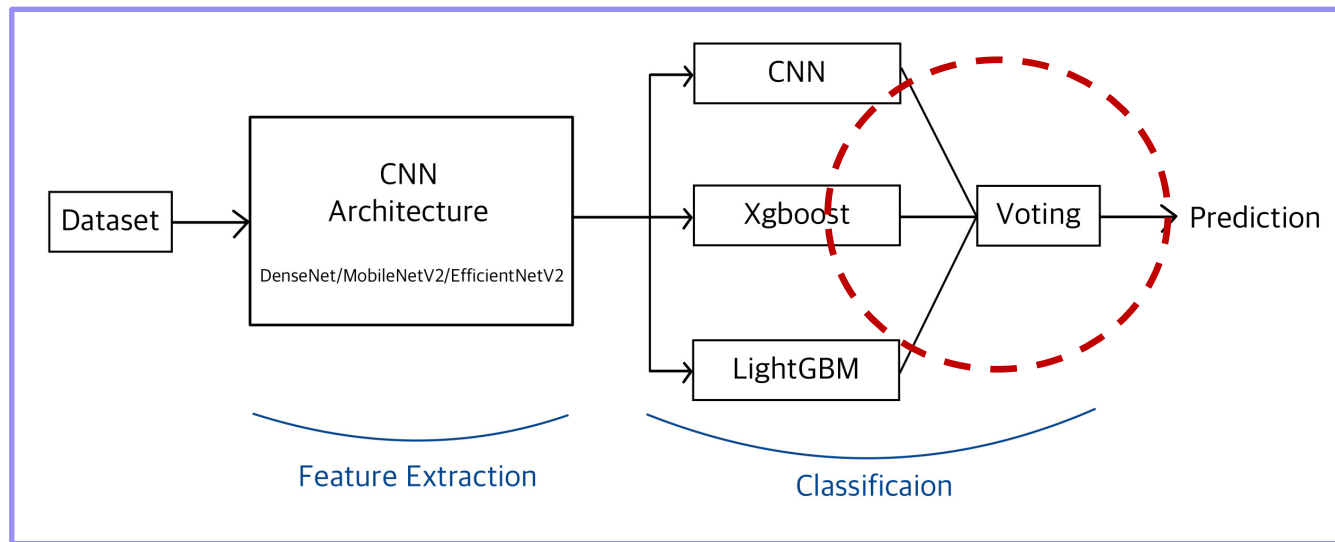
### XGBoost

효율성을 높이는 여러 기능을 포함한 gradient boosting 기반의 앙상블 방법

### LightGBM

계산 과정을 줄여 학습 속도를 개선한 gradient boosting 기반의 앙상블 방법

## 2. Modeling 모델 구성 요소



### Stacking Ensemble

여러 모델들을 활용해서 예측 결과를 각각 도출한 뒤,  
그 예측 결과들을 결합해 최종 예측 결과를 만드는 방법.



# 3. Experiments Using DenseNet

Feature extraction by DenseNet

개별 모델 성능

DenseNet				
Accuracy	Confusion matrix		Pred	
			no	Yes
0.859	True	No	433	8
		yes	116	325

XGBoost				
Accuracy	Confusion matrix		Pred	
			no	Yes
0.993	True	No	438	3
		yes	3	438

LightGBM				
Accuracy	Confusion matrix		Pred	
			no	Yes
0.992	True	No	438	3
		yes	4	437

Stacking

Test accuracy: 0.993

Confusion matrix		Pred	
		no	Yes
True	No	438	3
	yes	3	438

DenseNet



# 3. Experiments Using MobileNetV2

Feature extraction by **MobileNetV2**

개별 모델 성능

MobileNetV2				
Accuracy	Confusion matrix		Pred	
			no	Yes
0.5	True	No	0	441
		yes	0	441

XGBoost				
Accuracy	Confusion matrix		Pred	
			no	Yes
0.984	True	No	434	7
		yes	6	434

LightGBM				
Accuracy	Confusion matrix		Pred	
			no	Yes
0.985	True	No	435	6
		yes	7	434

Stacking

Test accuracy: 0.985

Confusion matrix		Pred	
		no	Yes
True	No	434	7
	yes	6	435

MobileNetV2



# 3. Experiments

Using EfficientNetV2

Feature extraction by EfficientNetV2

개별 모델 성능

EfficientNetV2				
Accuracy	Confusion matrix		Pred	
			no	Yes
0.355	True	No	32	409
		yes	160	281

XGBoost				
Accuracy	Confusion matrix		Pred	
			no	Yes
0.864	True	No	397	44
		yes	76	365

LightGBM				
Accuracy	Confusion matrix		Pred	
			no	Yes
0.867	True	No	396	45
		yes	72	369

EfficientNetV2



Stacking

Test accuracy: 0.863

Confusion matrix		Pred	
		no	Yes
True	No	390	51
	yes	70	371

# 3. Experiments

## Stacking 결과 비교

### Stacking 결과 비교

DenseNet			
Test accuracy			
0.993			
Confusion matrix		Pred	
		no	Yes
True	No	438	3
	yes	3	438

MobileNetV2			
Test accuracy			
0.985			
Confusion matrix		Pred	
		no	Yes
True	No	434	7
	yes	6	435

EfficientNetV2			
Test accuracy			
0.863			
Confusion matrix		Pred	
		no	Yes
True	No	390	51
	yes	70	371

DenseNet을 기반으로 한 stacking이 가장 좋은 성능을 보였다.

## 4. Conclusion 결론 및 한계점

- 단일 모델의 결과를 확인하였을 때, CNN 모델을 사용하여 예측까지 진행한 case보다 CNN으로부터 얻은 feature extraction에 부스팅을 적용하여 예측한 case가 더 나은 성능을 보였다.
  - ▶ 선행 연구에서 사용한 ArcFace 등의 face detection 모델을 이용하여 예측을 시도해볼 수 있다.
  - ▶ CNN 모델링 과정에서 epoch을 늘려 학습을 더 길게 하여 예측을 시도해볼 수 있다.
  - ▶ XGBoost, LightGBM에 대한 gridsearch 과정을 추가하여 더 optimal한 파라미터를 찾는 과정을 시도해볼 수 있다.
- DenseNet based stacking이 가장 좋은 test accuracy를 얻었다.
  - ▶ 하나의 feature extraction에 대해 3개보다 더 많은 방법론으로 예측을 진행하여 3개보다 많은 예측값을 이용한 stacking을 시도해볼 수 있다.

**감사합니다.**