

Level1 P-Stage 마스크 착용 분류 대회

RecSys-01조

1. 프로젝트 개요

1-1. 프로젝트 주제

COVID-19의 확산으로 전 세계 사람들의 경제적, 생산적 활동이 제약이 가해지고 있다. COVID-19의 가장 큰 문제점은 전염력이 높다는 것인데 이를 예방하기 위해서는 마스크를 올바르게 착용함으로써 감염자의 침, 비말 등을 차단하는 것이 중요하다. 따라서 추가 인적자원 없이 카메라에 비친 얼굴 이미지만으로 마스크 착용 여부와 올바르게 착용했는지 자동으로 가려낼 시스템을 구축해 본다.

1-2. 프로젝트 구조

- 입력: 성별, 나이에 따른 마스크 착용/미착용/오착용 이미지
- 출력: 나이(20대 이하/30 ~ 50대/60대 이상), 성별, 마스크 착용 여부(착용/미착용/오착용)를 구분한 18개 class
- 구조
 - 사전 학습된 EfficientNet-b2, Convnext의 FC Layer를 변경한 Fine Tuning 시행
 - 이후 EfficientNet-b2, Convnext 각각의 결과를 합친 Ensemble 구조

2. 프로젝트 팀 구성 및 역할

- 소경학 (팀장) : 하이퍼 파라미터 설계, Model Ensemble 설계
- 임경연 (팀원) : Data Augmentation, Convnext 모델 제작에 기여함
- 하동준 (팀원) : EfficientNet (b2/b4/b7 성능 분석), Data Augmentation, mislabeled data 수정
- 허민영 (팀원) : Data Augmentation, EfficientNet b-2 모델 제작에 기여함
- 노건웅 (팀원) : 모델, 옵티마이저, loss, 하이퍼 파라미터 실험, GitHub Projects 사용 기여

3. 프로젝트 수행 절차 및 방법

3-1. 프로젝트 사전 기획 및 문제 정의

짧은 대회 기간을 감안하여 Python 라이브러리를 통해 pretrained-model 과 이의 가중치를 받아 주어진 task 에 맞게 모델을 재 학습 하기로 결정하였다. 주어진 이미지 중 얼굴 부분이 학습에 큰 영향을 미칠 것이라 가정했고 이에 따른 이미지 crop 등 데이터 가공이 필요할 것이라 판단하였다.

3-2. 프로젝트 수행 절차 및 과정

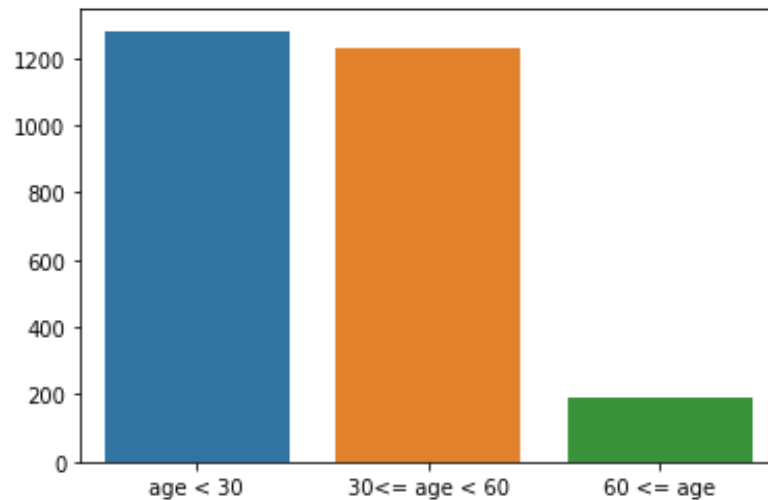
- 1주차(2/21~2/25)
 - (2/21~2/23) Baseline 코드 공개 전 개인 코드로 EDA, Dataset, Model, Train/Test진행
 - (2/23~2/25) Baseline 코드를 기준으로 Code Convention 통일 및 역할 분배
- 2주차(2/28~3/07)
 - (2/28~3/06) 성능 향상 기간으로, 세부 모델 선정, hyperparameter, scheduler 등 조정
 - (3/07) Stacking 등의 앙상블 기법을 적용 후 최종 결과물 도출

3-3. 탐색적 자료 분석 (EDA) 및 전처리

프로젝트 목표는 마스크를 잘 착용하고 있는지, 남성인지 여성인지, 나이대가 어떻게 되는지 분류를 하는 모델을 개발하

는 것이다. 따라서 위 세 항목에 대해서 분포를 확인해보자면, 아래와 같다

1. 나이: 나이는 총 3가지로 분류를 하는데 ① 30세 이하 ② 60세 미만 ③ 60세 이상이다. 전체 데이터 중 ①, ②는 동일하게 46%를 차지하지만 ③의 경우 8%를 차지한다.



위 데이터 분포를 살펴보자면 60세 이상 데이터가 지나치게 적어, 60세 이상 이미지에 대해서는 학습이 제대로 이루어지지 못할 확률이 높다. 따라서 Data Augmentation을 통해 데이터 증강시키거나, Oversampling 또는 Undersampling을 통해 Imbalanced Data에 대해 대응해야 한다.

3-4. 모델 평가 및 개선 과정

[EfficientNet-b2]

1. Backbone 모델 선정: EfficientNet-b2
2. 데이터 전처리 및 증강: 분류 목적과 관계 없는 배경에 집중하지 않도록 CenterCrop을 적용했지만 f1-score가 0.7265 -> 0.6690으로 감소하는 결과가 나와 얼굴이 테두리쪽으로 치우쳐있는 사진에는 좋지 못한 transform이라고 판단하였다. MTCNN을 활용하여 face-crop을 시도해보았지만 이것 또한 좋지 못한 결과를 얻었다. 따라서 Resize를 제외한 다른 transform은 적용하지 않았고, 60대 이상의 데이터 불균형 문제 해결을 위해 re-sampling을 적용한 결과 f1-score가 0.7265 -> 0.7319로 증가하였다.
3. 하이퍼파라미터 튜닝: batch_size = [16, 32, 64, 128], epoch = [10 ~ 30], optimizer = [Adam], lr = [0.001, 0.0001], lr_scheduler = [StepLR, CyclicLR, CosineAnnealingWarmUpRestarts, LambdaLR]을 적절히 조합하여 최적의 하이퍼파라미터 값을 찾는 과정을 수행했다.
과적합 문제와 acc 및 loss를 고려하여 batch_size = 32, epoch = 10~20, optimizer = Adam, lr = 0.0001, lr_scheduler = StepLR를 채택했다.

[Convnext Model]

1. Backbone 모델 선정: Convnext Large
2. 데이터 전처리 및 증강: 앞서 Imbalanced Data에 대응하기 위해 Data Augmentation을 진행했다. 전체 이미지를 flip left right, rotate, random erasing, random brightness를 통해 3배로 증강했고, 60대 이상 이미지는 7배로 증강했다. 또한 Dataset을 만들 때, CenterCrop을 통해 얼굴에 집중하도록 했으며, RandomPerspective, Resize를 통해 (224, 224) size로 Transform했다. 이후, Tensor로 변환 후 Normalize를 진행했다.
3. 하이퍼 파라미터 튜닝: batch_size = [16, 32], epoch = [5, 10, 20] loss_fuction = [cross_entropy, focal_loss]로 진행했다.

[Model Ensemble]

모델 Ensemble: 앞서 마스크, 성별, 나이 총 3개 모델을 만들어 Accuracy를 확인한 결과, 마스크와 성별의 Accuracy가 95% 이상 성능을 보여주었지만 나이는 약 75% 성능을 보여주었고, 그 중 60대 이상 이미지를 제대로 구분하지 못했다. 따라서, 모델 Ensemble은 상위 모델 3개를 사용했고, 기법은 voting을 사용했지만 만약 이미지 중 60대 이상으로 예측한 모델이 한 개라도 있으면 60대 이상으로 분류하도록 수정했다.

4. 프로젝트 수행 결과

model	Optimizer	Criterion	Batch_size	epoch	Pre-processing	LB 점수
EfficientNet-b2	Adam	CE	32	10	Resize (250, 190)	0.7265 77.9206
EfficientNet-b2	Adam	CE	32	12	Resize (250, 190) Re-sampling(추가)	0.7319 78.6349
Convnext	Adam	CE	32	5	Resize (224, 224) Centercrop, Randomperspective normalization	0.7218 77.8521

위에 나와 있는 모델 3개를 Model Ensemble 했을 때 f1 score는 0.7527, accuracy는 80.1905가 나오는 것을 확인할 수 있었다.

탐색적 자료 분석을 통해 60세 이상의 데이터가 부족한 것을 확인했고, 그에 따라 3개의 모델 중에서 한 개의 모델이라도 60세 이상이라고 판단했을 경우 그 판단이 옳을 확률이 높을 것이라는 가설을 세웠다. 이는 성별과 마스크 클래스가 일치할 때만 적용할 수 있도록 하였다. 예를 들어 model A와 model B는 마스크를 쓴 30세 이상 60세 미만 남성이라고 했고, model C는 마스크를 쓴 60세 이상 남성이라고 했을 경우 model C의 판단을 따르게끔 설계하였다.

그 결과 Ensemble을 한 모델의 f1 score는 약 0.02, accuracy는 2가 오른 것을 확인할 수 있었고 설계한 가설이 옳다는 것을 입증할 수 있었다..

5. 자체 평가 의견

· 하동준: Kaggle 형식의 데이터 대회에 처음 참가하여 한 사이클을 경험한 것과 프로젝트를 위한 협업을 진행한 것에 큰 의미를 둔다. TTA 나 Pseudo labeling 등의 다양한 앙상블 기법을 완성시키지 못한 것이 아쉽다.

· 허민영 : 대회 참여가 처음이어서 처음에 많이 서툴렀는데, baseline 코드를 전체적으로 훑어보고 실행시켜보면서 감을 많이 익힌 것 같다.

상대적으로 데이터가 부족한 카테고리에 resampling이 성능 향상에 좋은 작용을 한다는 것을 알게 되었으며 하이퍼 파라미터 또는 transform을 어떻게 적용시키느냐에 따라 적합한 pre-trained model도 천차만별로 바뀔 수 있다는 것을 알게 되었다. 더 많은 모델을 실험해보지 못한 것이 아쉽지만 추후에 많은 캐글 대회를 접해보며 적용해보고 싶다.

· 임경연 : 이번 프로젝트를 진행하면서 AI 프로젝트는 어떻게 업무 분장을 해야 되는지 이런 부분에서 잘 알지 못해 체계적으로 진행되지 못한 점이 아쉬웠다. 하지만 진행하면서 이때는 어디에 effort를 들여야 할 지, 감이 잡힌 것 같아 다음 프로젝트에서는 체계적으로 진행할 수 있을 것 같다. 그리고 AI 프로젝트에 관련된 많은 협업 방법론을 공부하려고 한다.

· 소경학 : 데이터 셋이 부족한 상황에서 모델을 설계하고 검증해 보는 경험을 해보면서 데이터 전처리가 굉장히 중요하다는 것을 깨닫게 되었고, 팀원들과 여러 의견을 공유했던 과정이 기억에 남습니다. 앞으로 남은 프로젝트에서 이번 프로젝트에서 깨닫게 된 내용들을 적용해 보고 싶습니다.

· 노건웅: 모델, 옵티마이저, 하이퍼 파라미터를 바꾸는 등의 다양한 실험을 진행하면서 강의에서 배웠던 여러가지 방법들에 대해 직접 실습할 수 있는 시간이라서 좋았습니다. 다음 번에는 데이터에 더 집중을 해보야겠다는 생각이 들었습니다.

이미지 분류 대회 개인 회고

RecOne T3245 임경연

1. 목표

이번 이미지 분류 대회 목표는 당연 1등하는 것이지만, 이번 대회에서는 처음하는 PJT이므로 각자 데이터 EDA, 전처리, 모델링 등 AI PJT 전체 과정을 스스로 한번 구현해보는 것에 있었습니다. 따라서, 우리는 각자 데이터를 전처리하고 모델링하면서 자기가 얻은 정보를 공유하면서, 얻은 정보를 자신의 모델링에 적용하고 이런 과정을 반복하면서 개인적인 성장을 위주로 프로젝트를 진행했습니다.

2. 모델 개선 과정 (개인)

- 데이터 양이 절대적으로 부족 → Pre-trained model 사용

이미지 분류대회를 위해 받은 데이터는 사람 2,700명, 이미지 18,900장으로 학습시키기 위한 데이터 양이 절대적으로 부족하다고 판단했습니다. 따라서, Pre-trained Model을 Fine-Tuning으로 진행했습니다. Pre-trained 된 모델을 사용하기 위해서 Timm 라이브러리를 사용했으며, Swin, Convnext, ViT, BEiT, Resnet, Efficientnet을 사용했습니다. 현재 이미지 분류에서 SOTA를 달성한 ViT, BEiT, Swin, Convnext 모델을 시도해봤으며, torchvision에서 성능이 좋다고 나온 Efficientnet, Resnet을 사용했습니다.

- 60대 이상 Age를 못 맞춤 → Data Augmentation을 통한 데이터 증강

이번 대회에서는 마스크, 성별, 나이를 분류하는 대회로, 저는 각각 마스크, 성별, 나이를 분류하는 모델을 나누어 학습하고 결과를 확인한 결과, 마스크, 성별은 Accuracy 95% 이상으로 분류를 잘하지만 나이는 80% 이하 성능을 보여주었습니다. 원인은 60대 이상 데이터가 다른 나이에 비해서 1/6 수준으로 확인하고 다른 나이대와 데이터양을 맞추기 위해서 Data Augmentation을 진행했습니다. 저는 Oversampling 또는 Undersampling 방법도 시도해보려고 했으나, 절대적 데이터 양이 부족하다고 판단하여 Data Augmentation을 진행했습니다. 좌우반전, Rotate, 밝기 조절을 통해 새로운 이미지로 인식할 수 있도록 변환했습니다. 이를 통해 데이터 증강을 하기 전과 비교하면 F1 Score 0.68 → 0.71 (▲0.3 (4.4%)) 상승하는 결과를 얻을 수 있었습니다.

- Model Ensemble

마지막 단계에서 모델 성능을 올릴 수 있는 방법은 모델 앙상블을 통해 여러 모델을 묶는 방법이라는 것을 찾았고, 적용해보았습니다. 앙상블은 voting, bagging, boosting, stacking이 있고, 과적합의 위험이 있지만 Kaggle 등 대회에서 모델 성능을 조금이라도 올리기 위해선 stacking 기법을 사용한다고 하여 stacking 기법을 시도해보았으나, 성능이 좋게 나오지 않아 경학 님이 voting으로 60대 이상 Weight를 주어 시도했고, 해당 기법이 성능이 더 좋게 나와 voting으로 선택했습니다. (Stacking: 0.71 / Voting: 0.75)

3. 개선할 점

- 체계적인 협업 진행 능력

이번 프로젝트에서 우리 팀은 U stage에서 배운 내용을 각자 직접 시도해보면서 학습하는 것에 중점이 맞춰져 있었습니다. 그래서 협업 과정은 서로 어떤 것을 했는지 결과가 어떻게 나왔는지 공유하는 과정이 전부였습니다. 또한, 저희 모두 AI PJT가 어떻게 진행이 되는지 잘 몰라 Task를 어떻게 나누고 분담해야 하는지 어려움이 있었습니다. 따라서, AI PJT 협업하는 방법에 대해서 더 많이 공부하고, 협업하기 위한 도구가 어떤 것이 있는지 알아보고, 다음 프로젝트에서 사용할 수 있도록 미리 적용해보려고 합니다.

- 모델링 능력

모든 개발자들에게 제일 중요한 능력으로 더 많이 개발 실력을 늘리려고 합니다. 기본 base code를 보면서 이해하는데 시간이 많이 걸렸고, 필요한 능력을 구현하는데 시간이 많이 걸렸습니다. 그래서 개발 실력을 길러 원하는 것을 나만의 AI Template을 만들어 팀원들에게 공유하고 바로 필요한 것을 구현할 수 있도록 하려고 합니다.

(허민영_T3235)

프로젝트 목표

- 주어지는 Baseline Code를 습득하여 내 것으로 만들고 최대한 많은 것을 실험해보기
- Jupyter 환경에서 벗어나 Python IDLE 환경에 익숙해지기, Github를 통한 협업에 익숙해지기

나는 내 학습목표를 달성하기 위해 무엇을 어떻게 했는가?

개인 학습 측면

- Special Mission을 통해 EDA, Dataset 및 Augmentation, 간단한 모델 구현, 다양한 기법을 적용한 training 코드 작성
- Jupyter 환경에서 벗어나 Pycharm을 활용한 IDLE 환경에서 실험
- Github 협업에 익숙해지기 위해 꾸준한 pull, push를 통해 반복 학습
- Batch_size, optimizer, loss_function, transform 등의 다양한 조합을 실험
- CenterCrop, MTCNN을 활용한 face-crop을 실험해보며 무엇이 더 좋은 결과를 만들 수 있을지 생각

공동 학습 측면

- 효율적인 협업을 위해 Baseline Code를 기반으로 학습
- Github를 활용하여 코드를 업로드 하며 협업하는 방법을 익힘
- 서로 질문이 있을 시 Slack과 Zoom을 활용하여 언제든지 도움을 주고받음
- 좋은 결과물을 종합해 앙상블

나는 어떤 방식으로 모델을 개선했는가?

- 사전 학습된 모델 EfficientNet b0, b2, b7을 가져와 실험해보며 최적의 모델 선정
- 데이터간의 불균형으로 60세 이상 나이를 잘 분류하지 못하는 문제를 해결하기 위해 resampling을 통해 데이터 증강
- 얼굴이 가장자리로 치우쳐져 있는 경우 Centercrop을 적용했을 때 얼굴이 잘리는 경우가 발생하였음
-> 이를 예방하기 위해서 MTCNN을 통해 얼굴을 감지하여 face-crop 하는 방법을 활용해 해결하고자 함
- 더 나은 신경망 학습을 위해 Learning rate를 0.001 -> 0.0001로 조정
- 학습률을 유동적으로 사용하기 위해 Learning rate Scheduling을 다양하게 적용해봄 (StepLR, CycLiLR, lambdaLR)

내가 한 행동의 결과로 어떤 지점을 달성하고, 어떠한 깨달음을 얻었는가?

- 새로운 Python IDLE 환경을 처음 사용해보며 Jupyter와 비교했을 때, 실험을 관리하고 실행하는 데 IDLE 환경이 더 편리하다는 사실을 깨닫게 되었음. 또한 시간적인 면에서도 훨씬 능률이 오른다는 것을 알게 되었음
- 팀원들과 협업하는 데 있어 큰 틀이 되는 Baseline Code의 중요성을 알게 되었고, Github를 통해 코드 변경 내용을 공유하는 것이 유용하다는 사실을 깨닫게 되었음
- 데이터 불균형의 문제가 모델의 성능에 큰 영향을 미치고 이를 해결하기 위한 방법으로는 데이터 증강이 매우 효과적이라는 사실을 알게 되었음
- Centercrop과 face-crop 모두를 실험해보았지만 모델의 성능에 크게 좋은 영향을 미치지 못했음
-> 이를 통해 이미지 데이터는 함부로 자르면 안된다는 유의미한 사실을 알게 되었음.

마주한 한계와 아쉬웠던 점은 무엇인가?

- 이론상과는 달리 성능이 좋아지지 않았던 적이 많아서 이미지 분류 모델의 정확도를 높이는데 있어 한계를 느꼈음
- Swin transformer, Resnet 등 다양한 모델 그리고 다양한 augmentation을 더 많이 적용해보지 못한 것이 아쉬움
- 계속되는 오류로 인해 tensorboard를 사용해보지 못한 것이 너무 아쉬움

한계/교훈을 바탕으로 다음 스테이지에서 새롭게 시도해볼 것은?

- 시작 전 전반적인 흐름에 대한 계획을 수립하고 시작할 것임. 또한 다양한 기법과 모델을 적용시켜볼 것임
- 어떤 사전 학습 모델을 선택하느냐에 따라 최적의 파라미터가 달라지기 때문에 이를 정확히 분석하고 실험할 것임

Wrap Up Report

노건웅_T3073

1. 이번 프로젝트에서 목표는 무엇이었는가?

이번 프로젝트에서의 목표는 약 5주동안 부스트캠프에서 배워온 지식들을 이용하여 대회에서 사용되는 실제 문제들에 적용해보고 활용해보는 것이 가장 큰 목표였다. 특히 중점에 둔 것은 2주차에 배운 PyTorch와 5주차에 배운 DL-Basic 지식들을 활용해보는 것이었다.

2. 나는 내 학습목표를 달성하기 위해 무엇을 어떻게 했는가?

- EfficientNet, ResNet, DeiT 등 다양한 모델을 사용
- 데이터를 직접 확인해보며 데이터간 불균형 확인
- SGD, Adam 등을 사용해보며 어떤 optimizer를 사용할 지 선택
- cross entropy, label smoothing loss, focal 등 loss를 다양하게 적용
- TorchVision의 Transform을 활용해보며 Data Augmentation 진행
- 그 외 CutMix, MixUp 등 여러가지 Data Augmentation을 적용해보려 노력함

3. 나는 어떤 방식으로 모델을 개선했는가?

- Pre-trained 모델의 FC Layer를 Fine-Tuning
- ColorJitter, CenterCrop, GaussianBlur, RandomErasing, CutMix 등 Data 수가 부족한 클래스에 대해 Data Augmentation 진행
- ResNet, DeiT, EfficientNet(b0 ~ b7)으로 점차 변경해보며 모델의 정확도 상승을 위해 다양한 실험 진행
- Label 간 구분이 명확하지 않은 경계값을 대상으로 label smoothing loss를 적용도 해보고 데이터가 불균형하므로 focal loss도 적용해보고 multi-class 분류이므로 cross entropy도 사용해보면서 어떤 loss가 더 성능을 올릴 수 있는지 다양한 실험 진행
- Momentum, SGD, Adam 3가지 optimizer를 대상으로 batch size, learning rate를 다양하게 바꿔보며 어떤 batch size, learning rate가 적합한지 실험
- 데이터간 불균형을 해소해보기 위해 60대 이상의 데이터들을 re-sampling해보므로써 기존과 어떤 차이가 있는 지 실험
- ResNet, EfficientNet-b7의 결과를 집계하여 Ensemble을 이용

4. 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떠한 깨달음을 얻었는가?

- 개인적으로 public data를 기준으로 높은 성능은 아니지만 f1-score 0.7이상을 달성해보았다. 한계 부분에서 적겠지만 결국에는 data를 더 자세히 분석해야한다는 것을 깨달았다.

5. 전과 비교해서, 내가 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?

- 이번이 처음 진행하는 프로젝트였으므로 전과 비교해서 새롭게 시도해봤다고 얘기할 수 있는 것은 없는 것 같다.

6. 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?

- 성능이 좋은 모델을 이용하면 점차 성적을 높일 수 있을 것이라 생각했지만 결국에는 어느 지점에서 멈춰졌고 결국에는 데이터를 깊게 분석해보지 못한 것이 한계였다.
- 가장 아쉬웠던 점은 다양한 transform 방법들을 적용해보았을 때 명확한 가설이나 이유없이 그냥 아무렇게나 진행해본 것이었다. 결국에는 모델의 성능을 끌어올리지 못하는 데이터들만 여러개 생산되어서 아무 쓸모 없는 data augmentation이 되었다.
- 결국 마스크 데이터를 구분하는 것은 얼굴인데 Face Detection을 이용해보지못한 것도 아쉽다.
- Data Augmentation이외에도 명확한 이유나 근거 또는 가설 설정 없이 행동이나 실험을 취해서 시간을 많이 소비했다는 것도 아쉬웠다.
- 마지막으로 급한 마음에 서둘러서 Validation Set에 대한 제대로 된 결과 분석없이 실험을 진행해본 것도 아쉬웠다.

7. 한계/교훈을 바탕으로 다음 P-Stage에서 스스로 새롭게 시도해볼 것은 무엇일까?

- Data EDA에 시간을 많이 투자할 것이다.
- 명확한 문제 정의를 바탕으로 근거와 이유 아래 다양한 가설들을 설정해보고 실험을 진행하고 결과 분석을 진행할 것이다.
- 무작정 Data Augmentation을 진행하는 것이 아닌 기본 데이터들을 이용해 충분히 실험해보고 부족한 부분에 대해서 Augmentation을 진행할 것이다.
- Model, Optimizer, Loss 등을 선택하는 것도 이리저리 바꿔보며 선택하는 것이 아닌 특정 가설이나 근거 아래 선택하기 위해 노력할 것이다.

Wrap Up Report

소경학_T3110

1. 이번 프로젝트에서 목표는 무엇이었는가?

이번 프로젝트의 목표는 모델의 성능을 높이기 위한 방법들을 연구함에 있었다. 데이터 전처리부터 하이퍼 파라미터까지 EDA를 통해 얻은 정보들을 토대로 다양한 방법들을 시도해보고 설계해보는 것이 주된 목표였다. 학습한 내용을 토대로 모델을 A부터 Z까지 설계하는 과정을 이해하는 것에도 목적이 있었다.

2. 나는 내 학습목표를 달성하기 위해 무엇을 어떻게 했는가?

- 하이퍼 파라미터를 다양하게 바꿔보며 최적의 성능을 내는 수치를 찾아봄
- Data augmentation을 transform을 다양하게 사용해보며 진행함
- 다양한 모델을 앙상블을 해보고 결과를 기록하고 분석함
- EDA를 통해 문제 정의에 초점을 두고 목적에 맞게 모델을 설계함

3. 나는 어떤 방식으로 모델을 개선했는가?

- 60세 이상의 데이터를 늘려보며 성능을 개선하였고, 다양한 transform을 써보면서 데이터 augmentation을 하였다.
- 높은 성능을 기록한 모델 (efficientNet-b2, efficientNet-b7, Resnet50)을 앙상블을 통해 성능을 개선함
- 여러 모델의 결과를 토대로 성능을 분석해보고 라벨에 따른 앙상블을 시도함
- batch size와 learning rate를 조절해보면서 최고의 효율을 낼 수 있는 값을 찾아봄
- 이미지의 나이를 변경해보면서 60대 이상의 데이터들을 효과적으로 늘릴 수 있는 방법을 찾아봄

4. 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떠한 깨달음을 얻었는가?

- efficientNet-b2와 Resnet을 앙상블을 통해 f1 score로 0.7580을 달성하였고, accuracy는 약 80%를 달성하였다. 모델을 설계하기 전에 데이터를 분석하는 과정이 무엇보다 중요한 것을 깨달았다. 데이터 전처리를 할 때에도 우리가 목표로 하는 것이 무엇인지 확실히 인지하고 했을 때 더욱 효과적인 방법으로 성능을 개선할 수 있는 것을 알게 되었고, 시간 또한 절약할 수 있다는 것을 알게 되었다.

5. 전과 비교해서, 내가 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?

- 문제 정의에 좀 더 시간을 투자했다. 문제 정의를 하지 않고 접근하였을 때 보다 실험을 하는데 있어서 목적이 뚜렷하게 생겼고 실험을 통한 결과를 분석할 때도 좀 더 효과적으로 할 수 있게 되었다.

6. 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?

- 모델간의 차이를 뚜렷하게 밝혀내지 못한 것이 아쉬웠다. 제출횟수에 한계가 있어서 다양한 모델을 여러 경우의 수로 나누어 실험하지 못한 부분이 아쉬웠다.
- epoch에 따른 성능 차이를 뚜렷하게 구분하지 못한 것이 아쉬웠다. 하이퍼 파라미터와 epoch의 상관관계를 좀 더 명확히 밝혔으면 좋았을 것 같다.
- CutMix를 통해 모델의 성능을 개선시키지 못한 점이 아쉬웠다.
- 현재 SOTA 모델을 실력 부족으로 구현하지 못하였고, 그렇기에 그 모델로 실험을 못 해본 것이 아쉬웠다.

7. 한계/교훈을 바탕으로 다음 P-Stage에서 스스로 새롭게 시도해볼 것은 무엇일까?

- 문제를 올바르게 정의하고 진행할 방향을 정하는데 시간을 더욱 투자해보고 싶다.
- pretrained된 모델보다 문제에 걸맞는 방향으로 모델을 직접 설계해보고 싶다.
- 데이터 augmentation시 실험에 대한 결과를 명확하게 기록하고 차이점을 정확하게 분석하며 진행해 보겠다.
- 베이스라인 코드를 명확하게 분석해보고 코드를 다양하게 바꿔가며 내 것으로 만들어 보고 싶다.

Level 1 P Stage 개인 회고

하동준

프로젝트 목표 설정

'첫 데이터 대회 만큼 그 과정에 온전히 참여하고, 데이터, 모델, train/test 의 한 사이클을 경험 하자' 가 일차적인 목표였고 부수적으로 성적 또한 만족스럽게 나온다면 좋겠다는 생각을 했다.

EDA 및 문제인식

- gender / race / age 를 조합하여 총 18 개의 classes
- (height=512, width=384) 이미지를 입력 받아 class 예측 하는 image classification
- 총 2700 명, 인 당 7 장(정상 착용 5 장 / 미착용 1 장 / 잘못 착용 1 장), 총 18900 장
- male 1042 명, female 1658 명
- 데이터가 ≤ 20 과 50 ~ 60 에 밀집되어 있음
- age 라벨 중 60 대 이상을 가르키는 라벨의 데이터 수(class 2, 5, 8, 11, 14, 17)가 현저히 적다
- 전체적으로 마스크를 착용하고 연령이 60 미만인 class 0, 1, 3, 4 가 많은 imbalanced data 임

Data

- 실제 이미지와 이미지가 가르키는 label 이 잘못된 경우가 다수 존재하여 이를 수정 필요
- 토론게시판 참고
- 일차적으로 전수조사 진행한 캠퍼의 게시글 통해 수정
- 이후 토론 게시판 글을 이용해 수정 및 데이터셋 저장
- 초기 baseline code 이용 이전에는 단순히 모든 사진들을 8:2 로 train/val split 하였음
- 하지만 같은 인물의 사진이 train 과 val 에 섞여 들어가며, val acc/f1 test 값이 비정상적으로 높아지는 결과발생
- 따라서 인물별 사진을 저장하는 폴더들을 8:2 split 후 개별 dataset 생성

Model & Train

- 초반에는 Baseline Code 에 정의 되어 있던 기본 CNN based model 을 정의하고 이를 학습 후 label 예측
- 이후 efficient net 의 b2~b7 까지의 모델을 pretrained 가중치와 함께 가져와 마스크 이미지 분류 task 에 맞게 학습 진행
- 가벼운 b2 부터 무거운 b7 까지 b7 이 10 epoch 도는 시간 동안의 최대 epoch 으로 학습 후 f1/acc 비교 -> 큰 차이는 없다.
- augmentation 진행 (centercrop/colorjitter/RandomRot/RandomHorizFlip/RandomPerspective/Normalize)
- 학습에 얼굴이 가장 중요할것이라 판단하여 centercrop 사용. 크기가 충분히 크지 않으면 얼굴이 잘리는 경우도 발생. 결과가 크게 좋지 않았음. 따라서 얼굴만 crop 한다가보다는 네 모서리들을 약간 잘라낸다는 느낌으로 크기를 늘린 centercrop 진행
- colorjitter 로 이미지의 다양한 변화. Random 시리즈들로 정면외에 기울어진 얼굴들을 학습시키려 노력
- MTCNN 으로 얼굴 감지하고 crop 하여 새로운 이미지들을 생성.
- *opencv* 이용 얼굴 추출도 해보자
- 60 대의 이미지 및 그 외 이미지들의 augmentation 을 위해 같은 이미지의 사진들을 추가 생성하여 transforms 하였지만 좋은 결과를 얻지 못함.
- 따라서 60 대의 사진만 한번 더 resampling 하기로 결정
- focal 이 imbalanced dataset 을 학습 시키는데 효율적이라는 말을 듣고 cross-entropy -> focal 변경

- eval 을 위한 이미지들을 제외한 이미지들을 기존 train/val 8:2 split 에서 자체 f1/acc 테스트를 위하여 train/val/test 6:2:2 split
- scheduler 교체 & lr 변경. cyclelr 사용
- stacking ensembling 으로 상위 결과 도출한 모델 3 가지의 예측 csv 를 수정
- age 데이터의 부족으로 gender/mask 는 90 이상의 정확도를 갖지만 age 는 70 근처의 성능.
- 전체 f1/acc 값은 age 의 값과 거의 동일
- 60 대 데이터 부족. 따라서 60 대라고 예측한 것은 부족한 데이터에도 60 대라고 한 것이기 때문에 다른 것보다 확신이 있다고 생각. 따라서 세 모델 중 하나라도 60 대라고 예측하면 60 대로 수정. 하지만 gender 나 mask 예측값 까지 다르다면 잘못된 예측이라 가정하고 무시

회고 및 다음 대회에 대한 다짐

결과에 대한 원인 분석 및 가설 검증이 부족했다고 생각하며, 시간 부족으로 인해 TestTimeAugment, PseudoLabeling 등의 앙상블 기법을 완성 시키지 못했던 점, 그리고 더 좋은 성능을 내지 못했던 점이 아쉽다. 데이터 대회에 흥미와 자신감을 얻었던 것 같고 추진력을 받아 다음 대회에는 더 계획적으로 더 좋은 결과를 낼 자신이 있다.