

재활용 품목 분류를 위한 Semantic Segmentation

7조 - 컴퓨터 구조

1. 프로젝트 개요

- 프로젝트 주제 및 개요
: 재활용 쓰레기 Semantic segmentation
- 기대 효과
 - 쓰레기장에 설치되어 정확한 분리수거 도와 환경 부담 줄이기
 - 어린아이들의 분리수거 교육에 활용

2. 프로젝트 팀 구성 및 역할

- 고재욱 : hr-net high epoch 실험
- 김성민 : mmdetection lib, mmsegmentation lib 구성, visualization tool base, crf, upernet-swin, HrNet 실험, ensemble, obj-augmentation debugging, pseudo labeling, mmsegmentation에 albumentation 적용
- 박지민 : BiSeNet, ISANet, EMANet 실험
- 박진형 : mmsegmentation lib 구성, OCR-Net 실험, HrNet 실험, wandb Sweep
- 심세령 : confusion matrix, smp lib 활용한 모델 종류 실험, UneXt 실험, Loss weight hyperparameter tuning
- 윤하정 : obj-augmentation, visualization tool upgrade, smp lib 활용한 model 실험, augmentation 실험

3. 프로젝트 수행 절차 및 방법

기간	활동 내용
2021.10.18 ~ 2021.10.22	<ul style="list-style-type: none">○ Library(mmsegmentation, torchvision, smp, ...) 환경 구축○ 모델 선정○ visualization, crf 구현
2021.10.25 ~ 2021.10.29	<ul style="list-style-type: none">○ 시각화, complex matrix를 통한 성능 개선○ 성능 개선을 위한 자료 찾기○ augmentation 환경 구축

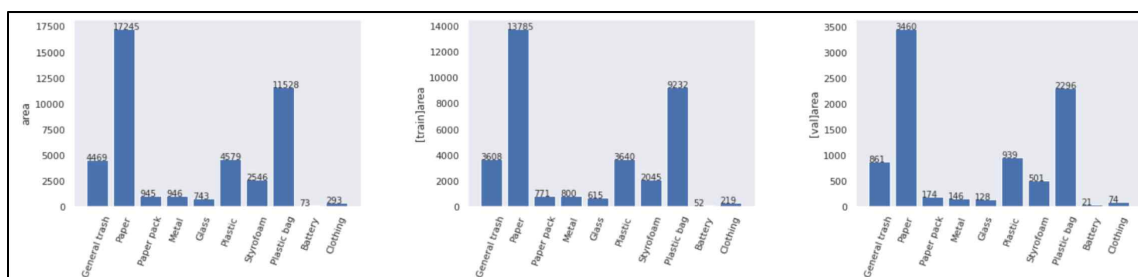
2021.11.01 ~ 2021.11.04

- Ensemble
- Object augmentation
- TTA
- psuedo labeling

4. 프로젝트 수행 결과

● 탐색적 분석 및 전처리

- 학습데이터: Train 2617장, Valid 655장
- Input: (512, 512, 3) 이미지
- Output: Pixel 좌표에 따른 카테고리 값



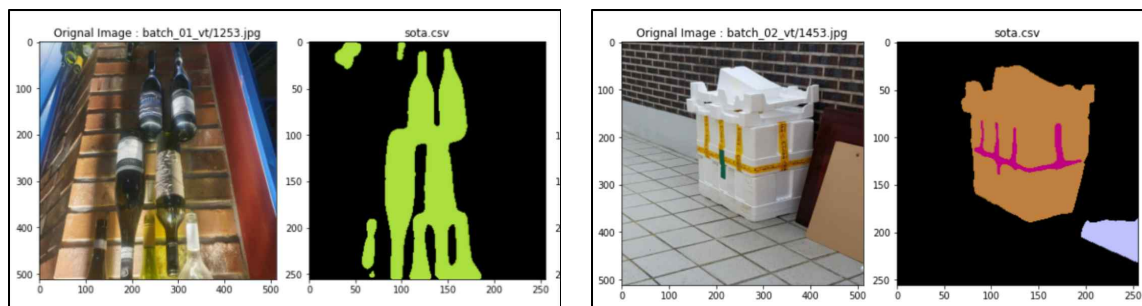
● 모델 선정 및 실험

모델	모델 선정 이유 및 개선 방법
HrNet	다양한 resolution을 학습함으로써 성능의 개선을 기대함 - OCR decoder 달아보기
upernet_swin	다양한 계층을 활용할 수 있는 구조, 엔지니어링의 용이성, 준수한 성 - 시각화, validation을 통해 수정 방향 설정 & augmentation, TTA, decoder 및 auxiliary head 수정 등을 통해 모델 개선
BiSeNet	경량화로 빠른 학습 속도를 가짐으로써 다양한 방향의 실험을 기대함 - input size 맞춰주기(cityscape비율 2:1을 1:1로), epoch 수 늘려주기, validation 적용, auxiliary head 수정
DeepLab	receptive field, 깊은 구조 등을 통한 성능 상향을 기대함 - resize, rotation, blur, gaussian blur 등 augmentation 실험
OCRNet	Backbone모델 swin으로 변경 함으로써 성능 개선
UneXt	Unet과 같이 의료 분야 데이터(적은 데이터)에 적합한 모델 - Backbone 변경, augmentation 적용, encoder 변경

● 모델 개요

- Architecture: **UperNet + Swin transformer**
- LB Score: 0.753(Public) / 0.720(Private)
- Optimizer: AdamW - Lr = 0.00006, betas=(0.9, 0.999), weight_decay=0.01
- Scheduler: CosineAnnealing - warmup='linear', warmup_iters=300, warmup_ratio=0.001, min_lr_ratio=1e-4
- Training time augmentation
 - Blur, rgb shift, HueSaturationValue, multiscale, RandomFlip, RandomCrop, PhotoMetric Distortion
- Test time augmentation
 - multiscale, RandomFlip
- 추가 시도
 - Object augmentation
 - Pseudo labeling

● 시연 결과



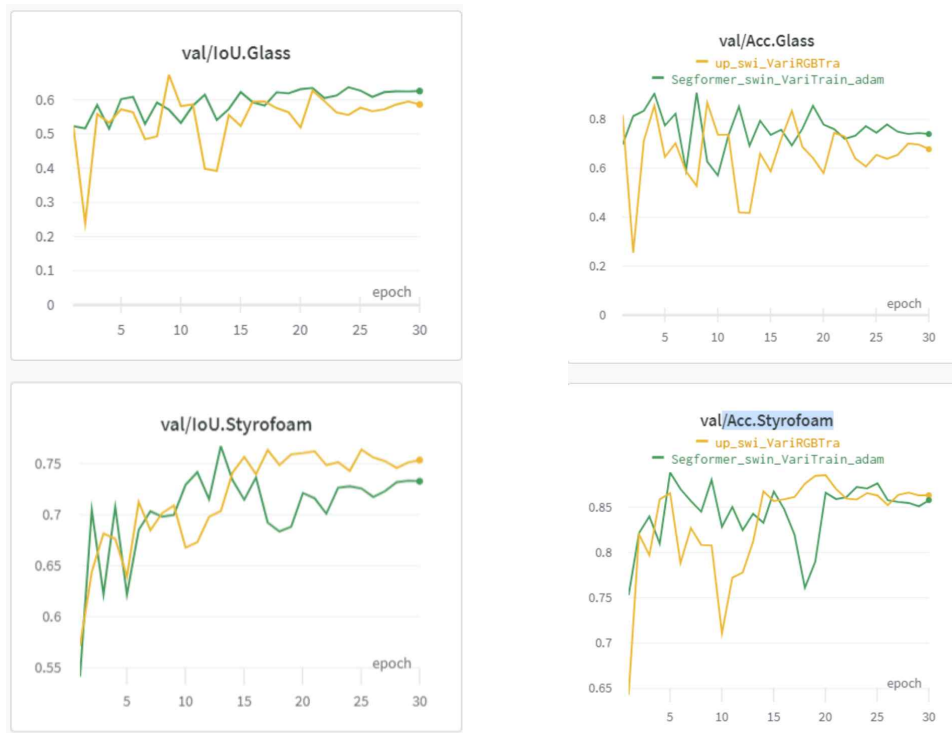
5. 앙상블 전략

valid를 살펴보면 모델별로 더 잘찾는 라벨이 달랐음 -> 이를 ensemble에 활용

● 1번 ensemble way : glass의 케이스를 살펴보면 acc, iou모두가 비례하게 높은 것을 확인할 수 있음, 즉 초록색 모델이 glass의 위치를 더 잘찾았다는 결론이 나옴 초록색 노란색을 base로 깔고 초록색 모델이 glass로 예측한 부분은 glass로 라벨링

● 2번 ensemble way : styrofoam의 케이스를 살펴보면 acc는 거의 비슷한데 iou면에서 차이가 있는 것을 살펴볼 수 있음 즉, 스티로폼이 아닌 곳에도 스티로폼으로 라벨링을 해주었기 때문에 iou가 더 낮은 것으로 판별

초록색 모델이 스티로폼이라고 예측했지만 노란색이 스티로폼으로 예측하지 않은경우 노란색 모델이 예측한 라벨로 대체, 이경우 단점은 노란색이 예측한 라벨로 대체하는 과정에서 대체한 부분도 예측을 잘못했다면 그 라벨의 iou가 떨어질 수 있긴 했었음



6. 자체 평가 의견

- 잘한 점들
 - 학습된 모델 결과를 시각화하여 문제점 파악
 - 다양한 기능들을 실험함(augmentation, psuedo labeling등)
 - validation 자료를 좀더 효율적으로 사용함(confusion matrix)
- 시도 했으나 잘 되지 않았던 것들
 - Class imbalance를 해결하기 위해 loss function 변경했지만 효과를 보지 못함
 - Hyperparameter tuning
 - decoder, auxiliary_head의 변환
 - CRF
 - RGB shift, blur, image compression TTA
- 아쉬웠던 점들
 - K-fold strategy
 - Swin transformer large
 - 기능을 적용했을 때에
 - 모델을 많이 돌려보지 못함
 - ensemble을 단순히 valid를 보고 삽입하는 형태가 아닌 hard voting이나 soft voting에 이를 섞어줬다면 더 좋을듯함 (ex - hardvoting을 하되 유독 glass를 잘 맞추던 모델이 있다면 그 모델이 glass를 예측했을 경우 glass로 라벨링)