

Semantic Segmentation for Recycling Trash Data

1. 프로젝트 개요

- 프로젝트 목표
 - 재활용 이미지 semantic segmentation 모델 개발
- 데이터셋
 - COCO format 이미지 3272장 (512x512 resolution)
 - Class : Background, General trash, Paper, Paper pack, Metal, Glass, Plastic, Styrofoam, Plastic bag, Battery, Clothing
- Tool: Pytorch, mmsegmentation, Weights and Biases, Github, Notion

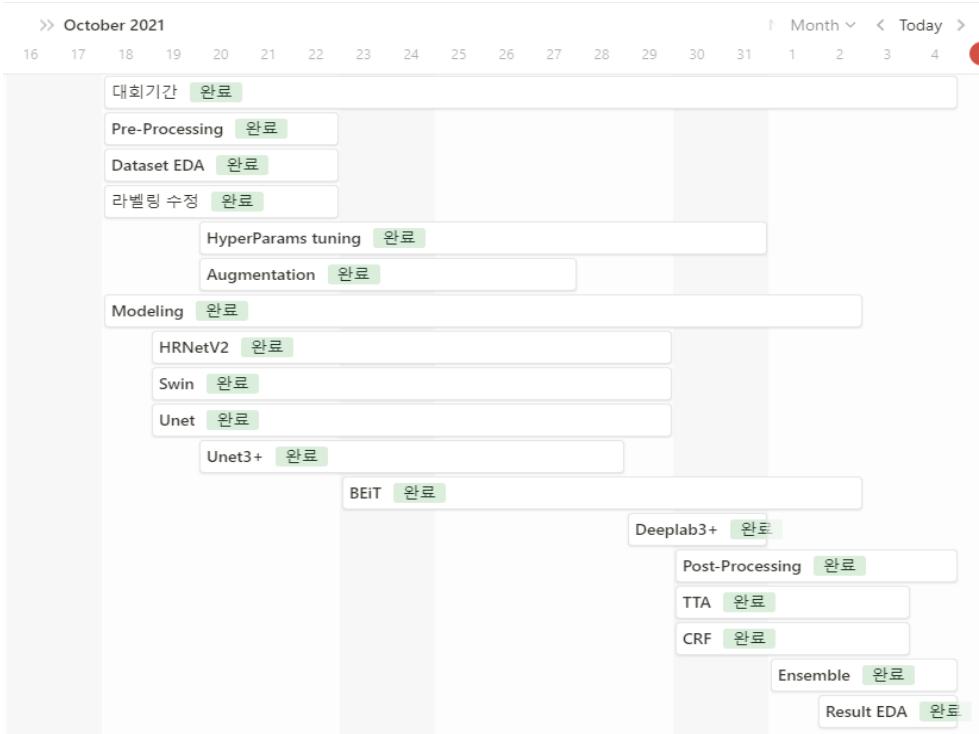


2. 프로젝트 팀 구성 및 역할

Name	Role	Description
강재현	Modeling Pseudo Labeling Post-Processing Validation Split	Stratified Split을 활용한 validation set 도출 LB가 높은 제출 파일을 pseudo mask로 활용하여 Testset에 대한 학습 SwinTransformer, BEiT 학습 및 튜닝
김민준	Modeling Pseudo Labeling Post-Processing	Baseline code 작성 SMP 모델 중 Unet++ & DeepLabv3+를 이용하여 여러가지 loss 실험 pseudo labeling 적용하여 결과 도출, 이후 inference 단에서 TTA 적용
박상현	Visualization Modeling Post-Processing	Github과 DevOps 를 관리 Wandb 기능을 활용하여 EDA를 위한 Visualization 수행 UNet3+ 와 다양한 backbone의 mmseg UNet 실험
서광채	Modeling Ensemble Augmentation	ObjectAug를 응용하여 Background 영역에 Object 단위로 증강하는 PassionateMix 개발, BEiT 학습 및 튜닝, Pixel-wise Hard Voting 코드 작성
이승우	Modeling Visualization Post-Processing	Train image 및 학습결과 visualization 및 분석 HRNetV2+OCR 학습 및 튜닝 다양한 조합의 Augmentation 및 CRF 실험 및 적용
오하은	Inspection Modeling Pre-Processing	Train image 검수 학습결과 visualization 및 분석 BEiT 학습 및 튜닝

3. 프로젝트 수행 절차 및 방법

- 1주차(21.10.18~21.10.22) : 데이터셋 검수 후 잘못된 라벨링 수정/삭제, 자료 수집 및 EDA 수행
- 2주차(21.10.25~21.10.29) : 모델링 수행, Augmentation, Loss, Optimizer 실험
- 3주차(21.11.01~21.11.04) : 좋은 결과가 나온 모델들 선별, 이후 hard voting 양상별, 시각화를 이용한 최종 결과 출력



4. 프로젝트 수행 결과

- 모델 개요
 - BeIT(BERT Pre-Training of Image Transformers) : 이미지 패치를 토큰화하여 랜덤으로 mask를 씌워 학습하는 모델
 - Swin Transformer: Shifted window를 활용한 hierarchical transformer를 기반으로 하는 모델
 - UNetPlusPlus: Backbone을 통한 Down Sampling 후 Up Sampling을 거치는, U자 모양을 가진 모델
 - HRNetV2: High Resolution의 특징을 계속 유지하며 다양한 resolution의 특징을 병렬적으로 고려하는 모델
 - DeepLab: Encoder를 통해 Down Sampling, ASSPP를 적용 후 Decoder를 통해 Upsampling을 하는 모델
- 모델 선정 및 분석
 - BeIT (LB mIoU: 0.796)
 - Train Augmentation
 - Normalize, RandomFlip, AutoAugment, Multi scale = (256~1024, 256~1024), RandomCrop = (384, 384), PhotoMetricDistortion
 - Test Time Augmentation
 - Multi scale = (256~1024, 256~1024), RandomFlip, Normalize
 - Pseudo Labeling (2회)
 - Swin Transformer (LB mIoU: 0.792)
 - Train Augmentation
 - Normalize, Multiscale & RandomCrop, RandomFlip, PhotoMetricDistortion

- Test Time Augmentation
 - Multi scale = (256~1024, 256~1024), RandomFlip, Normalize
 - Pseudo Labeling (2회)
- UNet++ (LB mIoU: 0.742)
 - Train Augmentation
 - RandomResizeCrop(512,512), HorizontalFlip(p=0.5)
 - Test Time Augmentation
 - HorizontalFilp(), VerticalFlip(), Rotate90([0,90])
 - Pseudo Labeling (2회)
- HRNetV2 (LB mIoU: 0.705)
 - Train Augmentation
 - Rotate(limit=90), RandomResizedCrop(512,512)
 - Data augmentation with mix
 - Test Time Augmentation
 - Multi scale = (256~1024, 256~1024)
 - Pseudo Labeling (1회)
- DeepLabv3Plus (LB mIoU: 0.688)
 - Train Augmentation
 - RandomResizeCrop(512,512), HorizontalFlip(p=0.5)
 - Test Time Augmentation
 - HorizontalFilp(), VerticalFlip(), Rotate90([0,90])
 - Pseudo Labeling
- Augmentation:
 - Passion-ate Mix: ObjectAug를 응용하여 background 영역에 Object 단위로 증강하는 기법
- 검증 전략:
 - 데이터 검수, Human Error 제거, Stratified Hold Out
- 양상블:
 - Pixel-wise Hard Voting
 - Attention을 활용하는 Transformer 계열 모델과 local feature에 집중하는 CNN 모델을 양상블하여 향상된 성능 도출
- 대회 결과
 - Public / Private LB (mIoU) : 0.802 / 0.773
 - 최종 결과 : 1등 (19 teams)

1	Segmentation_16조	PA SS IO N- AT E🔥	0.802	115
---	------------------	-------------------	-------	-----

5. 자체 평가 의견

- 잘한 점들
 - 프로젝트 시작 단계에서 WBS 작성, 역할을 세분화하여 분업

- 이전 프로젝트 대비 다양한 모델을 두루 사용하여 진행
- Wandb를 활용하여 실험 분석 및 결과 비교를 효과적으로 수행
- 실험 내용을 팀 notion에 적극적으로 기입하여 결과 공유
- ObjectAug를 응용하여 자체 Augmentation인 Passionate-Mix 개발
- Hard Voting Ensemble 시도 및 성공
- 시도 했으나 잘 되지 않았던 것들
 - 시각화를 통해 hyper parameter을 조절하려 했지만, 사람의 눈과 컴퓨터 관점과의 차이 존재
 - BEiT에서 사용하는 mmsegmentation에 여러 가지 Loss를 추가하고 싶었으나, 워낙 큰 라이브러리이다 보니 성공적으로 외부 Loss function을 이식하지 못함
 - CRF의 hyper parameter tuning 최적화의 한계
 - Apex 사용 환경 구축에 어려움을 겪음
- 아쉬웠던 점들
 - FPN 계열 모델도 사용해봤다면 더욱 좋은 결과가 있을 것 같았다. 오래전 모델이라고 안써본 것이 아쉬웠음
 - Ensemble이 Hard Voting에만 그쳤는데 Soft Voting도 활용했으면 좋았을 것 같음

6. Appendix

- Inferred Images & Augmented Images



- Weights & Biases
 - <https://wandb.ai/passion-ate/segmentation/>
- Notion (Passion-ate)
 - <https://www.notion.so/kcseo25/Semantic-Segmentation-120bea3578c948f1b0da87b84bfc7ebf>