

Semantic segmentation wrap-up report

CV-9조 SCV

박상언 송지민 오왕택 이동호 이주현 지현동

1. 프로젝트 개요

1-1. 프로젝트 주제

뼈는 우리 몸의 구조화 기능에 중요한 영향을 미치기 때문에, 정확한 뼈 분할은 중요하다. 예를 들어, 뼈의 형태나 위치가 변형된 것을 객체 분할 모형을 통해 확인할 경우, 해당 문제를 빠르게 해결할 수 있다. 데이터는 손가락, 손등 그리고 팔이 촬영된 X-ray 데이터 셋이다.



라벨은 총 29 개로 각각의 라벨은 뼈 종류를 나타낸다. 우리는 사진 속에서 각 뼈의 종류들을 분할하는 작업을 수행해야 한다.

2. 프로젝트 팀 구성 및 역할

이름	역할
박상언	SegNet, 시각화
송지민	EDA, DeepLab v3 구현
오왕택	Pytorch lightning code, UNet++, Swinv2-UperNet 구현

이동호	Baseline code, FCN, DilatedNet, DeepLab v3+ 구현
이주현	증강 기법 실험, Git 관리, Git 템플릿 작성
지현동	Git 관리, 증강 기법 실험, U-Net 구현

3. 프로젝트 수행 절차 및 방법

3-1. Time-line

기간	수행한 일
2월 5일 ~ 2월 7일	강의 수강, 제공 데이터 및 코드 확인, baseline code 작성
2월 8일 ~ 2월 12일	설날 연휴, 휴식
2월 13일 ~ 2월 16일	데이터 살펴보기(EDA), 모형 실험
2월 17일 ~ 2월 21일	모형 실험, Git 정리

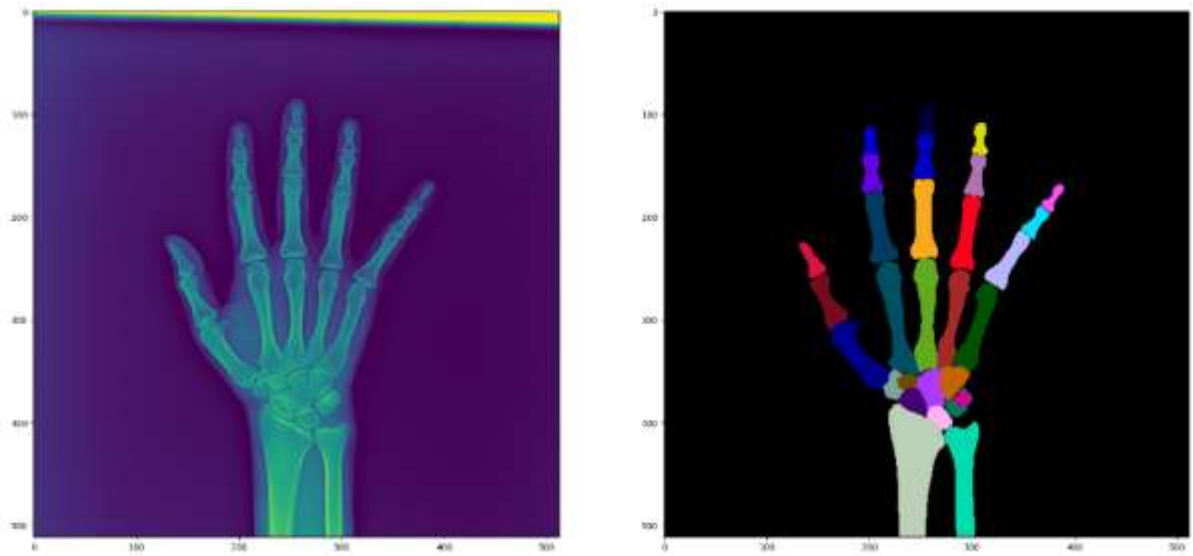
3-2. 프로젝트 수행 방법

1. 데이터 살펴보기 & EDA
2. 모형 학습

4. 프로젝트 수행 결과

4-1. 데이터 살펴보기 & EDA

데이터의 종류가 손가락, 손등 그리고 팔로 구성이 되어 있다. 그리고 팔목 부분의 사진들을 뼈의 개수가 적은 반면에, 손가락과 손등을 찍은 데이터에는 관절마다 라벨링이 되어 있어 뼈의 개수가 많았다. 우리는 해당 부분에서 손가락과 손등 사진을 분할하는 것이 어려울 것이라고 예상하였다.



또한, 사진의 종류를 왼손 사진과 오른손 사진으로도 분류를 할 수 있었다. 다만, 왼손 사진과 오른손 사진에서 특별하게 차이가 나는 부분은 없었다. 다만, 증강 기법 중 horizontal flip의 효과가 데이터 자체에 존재한다고 판단을 하였다.

데이터 불균형 또한 심하게 나타났다. Trapezium, Trapezoid 등 손가락과 팔목 사이에 있는 라벨의 개수는 많았던 반면에, 나머지 부분들은 라벨의 개수가 적었다.

4-2. 모형 학습

최종 프로젝트와 병행을 하였기 때문에 이번 프로젝트의 경우에는 다양한 모형을 사용해보기로 하였다. 그래서 강의에서 배운 합성곱 신경망 모형들을 기반으로 실험을 진행하였다. 옵티마이저는 AdamW를 사용하였으며, 손실 함수로는 멀티 라벨 분할 문제이기 때문에 BCELoss와 이번 대회의 평가지표인 Dice Score을 고려한 DiceLoss를 결합한 손실 함수를 새로 정의하여 사용하였다. 배치 크기 32로 총 100 에폭을 학습하였으며, 조기 종료를 설정하여 10번 연속으로 성능 개선이 없을 경우 학습을 중단하였다.

모형	Dice score
FCN-8s(ResNet101)	0.9412
U-Net	0.9565
U-Net++	0.9707
DeepLabv3	0.9297
DeepLabv3+	0.9643
Swinv2+UperNet (Public : 7위, Private : 6위)	0.9716

기본적으로 모든 모형이 일정 이상의 성능을 보여주었다. 합성곱 신경망 계열에서는 UNet++의 다이스 점수가 가장 높았고, 전체적으로 가장 높은 모형은 트랜스포머의 계열이면서 지역적인 정보 포착에 장점이 있는 스윈 트랜스포머였다. 스윈 트랜스포머는 window based self-attention을 통해서 기존의 ViT가 가지고 있었던 지역 정보 포착 문제를 해결함으로써 전역 정보와 지역 정보를 잘 포착하여 분할에서 좋은 성능을 거둔다고 알려진 모형이었다. 실제로 우리의 작업에서도 스윈 트랜스포머의 성능이 가장 좋게 나왔다.

5. 자체 평가 의견

5-1. 잘했던 점

- 제한적인 시간과 자원 속에서도 효율적으로 시간과 자원을 활용하여 어느 정도 성능을 기록하며 프로젝트를 마무리할 수 있었다.
- Baseline code를 빠르게 작성하여 모형 실험을 일찍 시작할 수 있었다.

5-2. 아쉬웠던 점

- 최종 프로젝트로 팀원 전원이 처음 해보는 분야를 선택하여 의미적 분할에 집중을 못한 부분이 아쉽다. 해당 부분은 스스로의 복습을 통해서 해결해야 할 것 같다.
- 멀티 모달의 모형 크기가 크다 보니 최종 프로젝트 모형을 서버에서 사용하면 의미적 분할 모형이 공간 부족으로 같이 실험할 수 없던 점이 크게 작용했다. 그래서 어느 한 쪽을 선택해야 하는 부분이 아쉬웠다

5-3. 프로젝트를 통해 배운 점

- 의미적 분할이 어떠한 작동 방식으로 이루어지는 지에 대해서 알 수 있었다
- 의료 데이터를 다뤄본 경험이 좋았던 것 같다.

개인 회고

박상언

이번 대회에서는 점수보다 학습과 기본기에 더 중점을 두었습니다. 강의를 통해 Semantic Segmentation의 개념과 Baseline code의 학습을 중점으로 진행하였으며 많은 모델을 사용하기 보다 모델의 구조를 학습하고 코드로 작성된 모델을 확인하여 기본기를 다지는데 중점을 두었습니다. 이를 통해 기존에 학습한 FCN, Unet에 더해 DeconvNet, DeepLab, DilatedNet, Unet++ 등 다양한 모델의 주요 아이디어와 구현 방향, 한계점, 개선점을 학습할 수 있었습니다. 강의에 포함된 모델 구현 미션과 VGG 레이어의 크기 변경, Unet 성능 비교 등 학습에 중점을 두었으며 팀원이 작성해준 Baseline코드에서 SegNet의 성능을 확인하고 코드를 수정하였습니다. 이 과정에서 데이터 시각화를 활용하여 모델의 학습 과정을 다시 한번 확인할 수 있었습니다. 또한 팀 내에서 선정한 Git 컨벤션에 따라 이슈를 생성하고 Branch를 생성하여 commit후 merge하는 일련의 과정을 수행하였습니다. Git 활용은 계속해서 실습하고 공부하였으나 이번 프로젝트에서 다시 한번 Git Flow를 경험하며 최종 프로젝트에서 매끄럽게 협업하기 위한 Git활용 방법을 다시 한번 확인할 수 있었습니다. 다만 최종 프로젝트에서 사용할 생성모형과 backend, frontend를 같이 공부하며 대회를 진행한 탓에 Baseline code 작성과 모델 실험, 대회 성적면에서 아쉬운 점이 있었습니다. 대회 이후에도 추가적인 학습을 통해 Semantic Segmentation의 모델을 한번 더 확인하고 직접 작성해보는 시간을 가질 것입니다.

송지민

FCN, DeepLab, Unet 등을 이용해 semantic segmentation 하는 Baseline code가 제공되었습니다. 그 중 DeepLabV3를 맡아서 실험을 진행했습니다. 기본적인 코드로 테스트 했을 때 0.9297로 준수한 성능을 보여주었습니다. 또한 Baseline code에 Bottleneck, AtrousSpatialPyramidPooling과 같은 모델 구조가 정의되어 있어서 이해하기 쉬웠습니다.

400명의 사람 별로 왼손, 오른손 두 장의 이미지가 있고, 손가락, 손등, 팔이 촬영된 X-ray 데이터에 29개의 뼈 종류 라벨이 있습니다. 간단하게 EDA를 통해 데이터셋을 살펴봤는데 대부분 비슷한 위치에 뼈가 있고, 뼈의 크기 비율이 대체로 비슷했습니다. 평균적으로 손바닥에 모여있는 뼈들이 작았고 팔 부분의 뼈가 컸습니다.

각자 모델을 나눠서 팀 baseline code를 같이 작성했는데 그 과정에서 깃허브 양식에 맞춰 issue 작성 > 기능 개발 브랜치 생성 > 모델 개발 > 이슈 번호로 commit, push > pull request > merge 일련의 과정을 제대로 써본 것 같아서 다음 프로젝트에서는 더 잘 활용할 수 있을 것 같습니다.

아쉬운 점: Loss 함수 공부

다른 팀의 발표를 들어보니 여러 Loss 함수를 써보고 성능 비교를 했습니다. 그리고 Loss 함수를 여러 개 같이 쓰면서 비율을 조정해 최적의 성능을 찾았는데, 그 과정을 못해서 중요한 부분을 놓친 것 같습니다. 앞으로 자료를 찾아보면서 공부하고 여러 실험에 도전하면서 다양한 방법을 경험해야 한다는 것을 느꼈습니다.

오왕택

이번 Semantic Segmentation 프로젝트에는 점수를 올리는 것에 초점을 두지 않고, 코딩 실력 기르기와 모델 공부, 새로운 모델 사용해보는 것에 초점을 두고 진행하였습니다. 그리고 개인적인 목표로 과거에 Semantic Segmentation 프로젝트를 하면서 개인적으로 구현하지 못했던 SOTA 모형에 가까운 SwinTransformerV2 + UpperNet을 구현해보고 데이터를 살펴보는 것에 초점을 두었습니다.

우선 이번 대회를 진행하며 pytorch lightning으로 baseline 만드는 것을 목표로 하여 기존 baseline을 lightning으로 포팅하고 wandb나 smp 라이브러리, 디버그 등을 손쉽게 사용하도록 만들어 팀원들이 코드 작성에 시간을 아끼고 최종 프로젝트와 병행할 수 있도록 기여했습니다.

또한, Unet++모형을 smp 활용하여 모형 실험을 진행했고 PapersWithCode에 Semantic Segmentation 분야에서 높은 순위를 차지하는 모형을 직접 구현했습니다. SwinTransformerV2를 Encoder로 사용하고 UperNet을 Decoder로 사용하는 모형을 구현하여 모형 실험해봤으며 상세히는 사전 학습된 가중치의 불필요한 부분을 제거하고 특징맵을 가져와서 Decoder에 넣어주는 코드를 작성했습니다. 그렇게 본 모형을 사용하고 loss 실험, augmentation 실험으로 모형이 할 수 있는 최선의 성능을 끌어올리려 노력했지만 6등을 차지하였습니다.

최종 프로젝트의 주제를 지금까지 겪어보지 못했던 생성 모형을 공부하며 github에 다양한 코드 및 자료를 올리거나 일정 계획에 따라 체계적으로 진행하지 못한게 아쉬웠고, Semantic Segmentation은 자율주행, 의료 쪽에 많이 사용되므로 나중에 SegFormer 같은 트랜스포머를 공부하고 데이터 불균형 문제를 해결할 수 있는 방법들을 모색해보고 싶습니다.

이동호

석사 졸업 논문으로 의미적 분할을 연구하였기에 이번 프로젝트에서 제가 알고 있는 부분들을 많이 팀원들과 공유를 하고 싶었습니다. 다만, 최종 프로젝트 분야인 멀티 모달이 처음인 분들이 많아서 의미적 분할 관련 공부를 같이 못한게 아쉽습니다. 다만, 베이스라인 코드를 빨리 작성하여서 팀원들에게 공유하고, 여러 가지 모듈들을 설명해주어서 팀원들이 돌릴 수 있는 모형 세팅을 한 것은 좋았다고 생각합니다. 이미지 왜곡, CutMix와 같은 다양한 증강 기법을 적용하지 못한 점과 앙상블 기법을 적용하지 못한 점은 아쉽다고 생각합니다. 그리고 트랜스포머 기

반의 모형을 더 많이 사용해봤더라면 더 좋은 성능을 기록할 수 있지 않았을까 생각을 합니다. 그리고 이번에는 git pull request를 적극적으로 사용해본 경험이 좋았습니다.

이주헌

이번 Segmentation 프로젝트의 초반에는 강의를 들으면서 개인적인 공부에 집중했습니다. Segmentation의 개요와 모델들에 대해서 자세하게 알 수 있었고 첫번째로 FCN8, FCN16, FCN32 모델들을 Pytorch로 만들어보면서 기본적인 모델에 대한 공부를 진행했습니다. VGG16에서 FC6의 부분을 1x1 Convolution으로 표현했지만 7x7 Convolution으로도 교체해보는 미션을 진행하면서 그 이유에 대해서 생각하고 차원문제를 해결하면서 FCN과 segmentation에 대한 이해를 높였습니다. 다음으로는 여러 다른 모델들 DeconvNet / SegNet / DilatedNet / DeepLab v1 중 SegNet을 vgg16을 이용해서 구현해보는 미션을 수행했습니다. 이렇게 기본적인 모델들에 대해 알아보면서 segmentation에 대한 지식을 얻었습니다. 그 이후로는 Xception을 활용한 DeepLabV3+ 모델을 만들어보는 미션을 수행했습니다. 이렇게 강의와 미션을 수행하면서 개인적인 지식을 얻는데 집중하는 시간을 두번째주차의 초반까지 가졌습니다. 이런 과정을 거친후에 baseline 코드를 이용하여 학습을 진행하고 추론과 결과제출까지 해보았습니다. 아쉬운점은 다른 모델들은 학습 후 결과제출에 이용하지 않았고 이론적인 부분만 수행했다는점이 아쉬웠습니다. 추후에 개인적으로라도 다시 한번 공부 해 보고 싶습니다. 이후에 팀원들과 회의를 진행하면서 EDA 부분과 증강기법에 대해서 논의하고 수행하는 팀 활동 부분에서의 프로젝트를 진행하였습니다. 팀에서 할당된 증강기법 실험을 개인이 진행하고 데이터를 보면서 의견들을 나누었습니다. 또한 GITHUB에서 코드를 공유하며 진행했습니다. ISSUE, PR 템플릿을 만들고 기초적인 작업들을 하고 develop 브랜치에서 각자 branch를 만들어서 개인의 작업을 진행하고 develop으로 merge 하는 식으로 gitflow브랜치 전략으로 진행하였습니다. 사실 최종프로젝트에 집중하는 부분이 많았습니다. 그래서 github에 여러 자료들이 올라가진 않았고 잘 정돈되진 않았지만 최종프로젝트에 반영하기위한 github 관리의 연습이라고 생각하고 진행했습니다. 개인적으로 github에 대해서 공부를 하였고 팀원들에게 컨벤션과 관리방법에 대해서 알려주었고 최종프로젝트에는 더 잘 반영이 되도록 미래를 보며 수행하였습니다. Segmentation 부분에 대한 실험을 많이 진행하지 못한게 아쉬운 부분이지만 최종프로젝트와 병행하기에 어려움이 있었다고 생각합니다. 또한 이번 프로젝트를 진행하면서 기록은 기억보다 강하다 라는 문장이 기억에 남았습니다. 강의를 듣고 프로젝트에 참여를 했지만 강의에 대한 기억이 점점 사라지는 것을 느꼈습니다. 강의에 대한 기록을 꼭 해야한다는 것을 느끼면서 다음 프로젝트와 강의내용은 개인적으로 정리를 꼭 하려고 생각하고있습니다.

지현동

- Semantic Segmentation모형 공부

U-Net을 구현해보면서, semantic segmentation의 upsampling과정을 공부했습니다. FCN32 기초 모델부터 코드를 뜯어보면서 모델에 대한 이해도를 높였습니다.

- GitFlow 경험

풀리퀘스트를 진행할 때 발생한 conflict를 해결하는 일련의 과정을 경험할 수 있었습니다.
Develop 브랜치에서 feature 브랜치로 병합하면서, vscode의 편집기를 통해 conflict를 해결했고,
이후 다시 feature 브랜치를 Develop 브랜치에 squash & merge했습니다.