데이터 제작 프로젝트

학습 데이터 추가 및 수정을 통한 이미지 속 글자 검출 성능 개선 대회

cv-16조 비전 길잡이

1. 프로젝트 개요

OCR task는 글자 검출 (text detection), 글자 인식 (text recognition), 정렬기 (Serializer) 등의 모듈로 이루어져 있다. 본 프로젝트에서는 글자 검출 task에 초점을 맞추고, 모델을 고정한 상태로 데이터셋 추가 및 수정(Data Dirven Approach)을 통해 글자 검출 성능을 최대한으로 끌어올리고자 한다.

2. 프로젝트 팀 구성 및 역할

- 민기 : AI_hub dataset 탐색 및 적용, SynthTextKR 생성, SynthTextKR_mix finetune
- 박민지 : ICDAR 데이터 실험, SynthText sampling data 학습 후 fine-tuning 실험
- 유영준 : ICDAR 데이터 실험, Augmentation, SynthText pretrained 기반 fine-tuning 실험
- 장지훈 : ICDAR 데이터 실험 및 학습, pre-trained 정보 fine-tuning 실험
- 최동혁 : SynthText .mat 변환 및 500k en dataset 학습, ICDAR 17/19 fine-tuning

3. 프로젝트 수행 절차 및 방법

3-1. Model

본 프로젝트는 data-driven approach를 통한 글자 검출 성능 향상을 목표로 하기 때문에, 모델을 처음부터 고정시켜서 사용했다. 사용한 모델은 VGG-16을 backbone으로 사용한 EAST 모델로, text가 기울어져 있거나 흐린 상태에서도 실시간으로 탐지할 수 있는 장점을 가진다.

3-2. Dataset

3-2-1. Boostcamp Data

- 처음으로 돌려본 데이터셋은 Boostcamp data이다. 그런데, labeling을 여러명이서 작업하다 보니 data annotation에 대한 일관성이 모호해서 해당 데이터에 대한 평가가 낮을 수밖에 없었다. 또한 왼쪽의 그림처럼 다각형으로 bbox 처리된 annotation에 대해서 학습이 되지 않는 문제가 발생하여 최소, 최대 크기로 수정하여 학습시켜야 했다. f1-score 기준으로 0.3202가 나와 다른 데이터셋으로 학습시키는 방향으로 가기로 했다.

3-2-2. AI Hub

- 야외실제 촬영 한글 이미지
  해당 데이터는 숫자 annotation이 없었고, bbox도 aistage 기준과 많이 다르게 크게 그려져 있으며, (x,y,w,h)로 표현된 좌표로 인해, 기울어진 텍스트에 대한 bbox 학습이 불가능했다.

- 공공행정문서 OCR
  상기했던 야외 데이터처럼 (x,y,w,h) 좌표계를 사용했으나 크게 기울어진 단어가 없기에 학습에 전혀 문제가 없을 것으로 판단, Pre-trained data로 사용하기로 결정했다.

3-2-3. ICDAR

- ICDAR 17 MLT

9개 언어로 구성된 OCR 데이터로, 총 9,000장의 데이터가 존재하는데, 이를 Training 7,200장, Validation 1,800장으로 나누어 사용하였다.

- ICDAR 19 MLT
  10개 언어로 구성된 OCR 데이터로, 총 10,000장의 데이터를 Training에 사용하였다.

3-2-4. SynthText Data

- Synthetic Pre-generated Dataset
  858,750장 중 542,706장을 학습에 사용하였다. Synthetic 데이터로만 학습 후 Pre-trained로 사용하였다. matlab 파일인 gt.mat에 이미지에 대한 정보가 저장되어 있었기 때문에 변환을 진행하였다.

- E2E-MLT DATA
  6개의 언어 중 한국어 데이터만 사용하였다. 한국어 40432장중 Background가 다른 이미지만 추출하여 총 5,452장을 사용하였다. 이후 Synthetic Pre-generated Dataset에서 7939장과 합쳐서 공개된 test dataset의 한/영 비율을 맞추었다.

- Synthetic KR generate
  Github에 존재하는 SynthText_kr이 python 2로 구현되어 있어 3로 변경후 합성 이미지 생성을 시도하였으나 대회 기간 내에 해결되지 않아 최종적으로 사용되지 못했다.

3-3. Data Augmentation

3-3-1. Baseline

- Baseline 코드로 1024 resize, 10도 rotate, 512 crop 및 color jitter와 normalize가 주어졌다.

3-3-2. Normalize

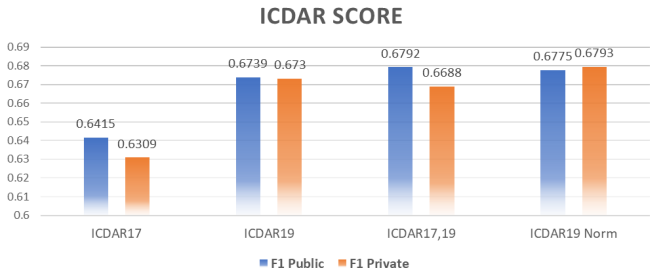
- Normalize 값을 이미지넷 기준 default 값으로 변경했을 때의 비교 실험에서 성능 향상이 있었다.

3-3-3. 그 외 Augmentation

- 이외에 Crop, Blur, Gaussian Noise, CLAHE, ToGray, ChannelShuffle 등 다양한 augmentation 기법을 적용하였는데, 대부분 성능이 좋지 않았다. 해당 augmentation 기법들이 글자 검출 자체를 방해한다고 판단하여, 적용하지 않았다. 최종적으로 3-3-1의 default augmentation에서 normalize 값만 바꾼 것으로 적용하여 실험을 진행했다.

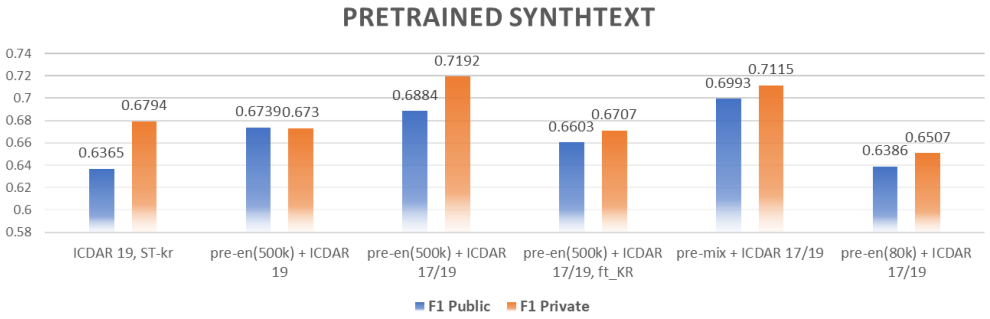
4. 프로젝트 수행 결과

우리 조는 OKR 방법론으로 글자 검출 챌린지에의 핵심 목표(objectives)와 해당 결과(key results)를 세워 달성하는 데 집중했다. 이번 챌린지에서 팀의 목표는 OCR 관련 이미지 외부 데이터셋을 10만장 이상 확보하여 충분한 학습을 시킨 후 f1-score 0.7 이상 만드는 것으로 설정하였고, 달성하였다.



<그림 1. ICDAR 데이터 모델 F1-Score>

ICDAR 데이터만 사용하여 학습을 시킨 결과는 그림 1과 같다. ICDAR19에서 Normalization 수정해줬을 때 점수가 약간 상승했고, ICDAR17과 ICDAR19 데이터를 학습에 다 사용했을 때 Public F1 점수가 가장 잘 나왔다.



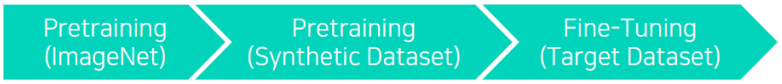
<그림 2. Synthetic Pre-trained 모델 F1-Score>

SynthText data를 학습 시킨 후 해당 pre-trained 모델을 통해 icdar dataset을 학습시켰을 때 나타난 결과는 그림 2와 같다. pre-mix는 한국어 5,452장과 영어 7,939장을 섞은 데이터셋을 학습시킨 경우에 대한 실험이다. ST_kr은 SynthText korean 데이터셋을 pre-trained에 사용하지 않고, fine-tuning 학습에 적용한 실험에 사용한 케이스이다.

본 실험 결과에서, pre-trained dataset을 통해 synthText 데이터에 대한 학습 효과가 더 좋게 나오는 것을 확인할 수 있었다. 특히 8만장 데이터셋을 pre-trained한 경우보다 50만장 pre-trained한 경우가 점수가 높게 나왔고, private에서는 pre-mix 보다는 pre-en 데이터셋에 대한 점수가 더 높게 나왔다.

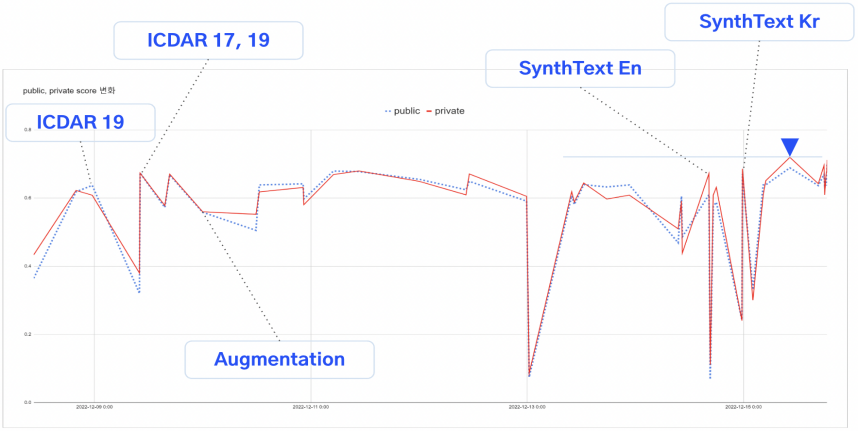
5. 자체 평가 의견

- 대량의 합성 데이터 이용해 pre-training을 미리 학습시켜서 fine-tuning을 진행하였다.



<그림 3. Training 순서>

- 대량의 SynthText 데이터에 대해 충분히 pre-training을 하지 못한 아쉬움이 있었다.
- 충분한 시간을 가지고 fine-tuning을 진행하지 못한 아쉬움이 있었다.
- 팀원들의 빠르고 신속한 팀워크로 대량 데이터 수집 과정의 이슈를 빠르게 확인하고 토의를 통해 결정 후 끝까지 학습한 결과 0.7192의 높은 점수를 기록할 수 있었다.



<그림 4. 최종 시간 별 F1-Score 그래프>