**3안. Detectron 활용**

**YOLO vs Detectron**

YOLO 특징 :

1. 빠른 디텍팅 시간 -> 실시간으로 객체 탐지가 가능하다.
2. 실시간 탐지 목적이 아닌 타 신경망에 비해 인식률이 떨어질 가능성이 크다.
3. C언어로 개발되어 있다.

Detectron 특징 :

1. Facebook Research에서 오픈소스로 공개한 Caffe2 기반의 프레임워크
2. Mask r-cnn을 이용해 객체의 모양에 맞춰 디텍팅 가능 -> 객체 면적 계산하기에 적합
3. Python으로 개발되어 있다.

**Detecron을 이용한 밀웜 학습 방법 및 결과**

1. COCO API json format에 맞춘 밀웜 데이터셋 생성(디텍트론은 COCO API를 사용하기 때문)
2. 밀웜 데이터셋 명을 detectron/datasets/dataset\_catalog.py 에 추가한다.

* dataset\_catalog.py : 사용 가능한 dataset 목록이 들어있는 파일

1. 만든 데이터셋으로 test 진행
2. cocoapi/PythonAPI/pycocotools/mask.py 의 area 함수를 이용해서, 디텍팅된 밀웜 및 배춧잎 면적 구함
3. 처음 디텍팅 된 배춧잎 면적 = A, 밀웜이 먹은 후 디텍팅 된 배춧잎 면적 = B 일 때

* (A – B) \* 100 / A = 밀웜이 먹은 퍼센트