import json

from tidecv import TIDE, Data

# Assuming cocoGtFile and cocoDtFile are defined paths to your JSON files

with open("/content/drive/MyDrive/P2M/dataset/val\_dataset.json") as f:

gt\_json = json.load(f)

with open("/content/drive/MyDrive/P2M/dataset/faster\_rcnn\_R\_50\_FPN\_3x/new\_data/coco\_instances\_results.json") as f:

dets\_json = json.load(f)

tide = TIDE()

# Initialize Data objects for ground truth and detections

gt\_data = Data('gt\_data')

det\_data = Data('det\_data')

# Process ground truth data

for det in gt\_json['annotations']:

image = det['image\_id']

\_cls = det['category\_id']

box = det['bbox'] if 'bbox' in det else None

mask = det['segmentation'] if 'segmentation' in det else None

gt\_data.add\_ground\_truth(image, \_cls, box, mask)

for det in dets\_json: # Directly iterate over dets\_json if it's a list

image = det['image\_id']

\_cls = det['category\_id']

score = det['score']

box = det['bbox'] # No need to check with 'if' since all detections have 'bbox'

# Add the detection information to det\_data

det\_data.add\_detection(image, \_cls, score, box)

# Evaluate based on bounding boxes

tide.evaluate(gt\_data, det\_data, mode=TIDE.BOX)

# Summarize and plot the results

tide.summarize()

tide.plot()