To calculate Map for the Yolo model:

from sklearn.metrics import average\_precision\_score

def compute\_map(y\_true, y\_pred, average="macro"):

"""Computes the mean average precision (mAP) from prediction scores.

Args:

y\_true: array-like of shape (n\_samples,) or (n\_samples, n\_classes)

True binary labels or binary label indicators.

y\_pred: array-like of shape (n\_samples,) or (n\_samples, n\_classes)

Target scores, can either be probability estimates of the positive

class, confidence values, or non-thresholded measure of decisions

(as returned by decision\_function on some classifiers).

average: {‘micro’, ‘samples’, ‘weighted’, ‘macro’} or None, default=’macro’

This determines the type of averaging performed on the data.

Returns:

average\_precision: float

The mean average precision score.

"""

return average\_precision\_score(y\_true, y\_pred, average=average)

Detection Algorithms Time Taken:

import pandas as pd

# Create a dictionary with the table data

data = {

"Object Detection Algorithms": ["Region-based CNN", "Fast RCNN", "Faster RCNN", "You Only Look Once"],

"Time(sec)": [62.7984, 54.9997, 47.4532, 7.2558]

}

# Create a pandas dataframe from the dictionary

df = pd.DataFrame(data)

# Print the dataframe

print(df)