Name – Vinayak A Belludi

Collage – Kle Technological University

DINO: Fine-Tuning and Evaluation for Custom Pedestrian Detection Dataset

1. Introduction

This report details the process and results of fine-tuning and evaluating DINO (DETR with Improved Queries) for a custom pedestrian detection dataset. DINO, an extension of the DETR (DEtection TRansformer) model, incorporates improved denoising anchor boxes for enhanced object detection performance.

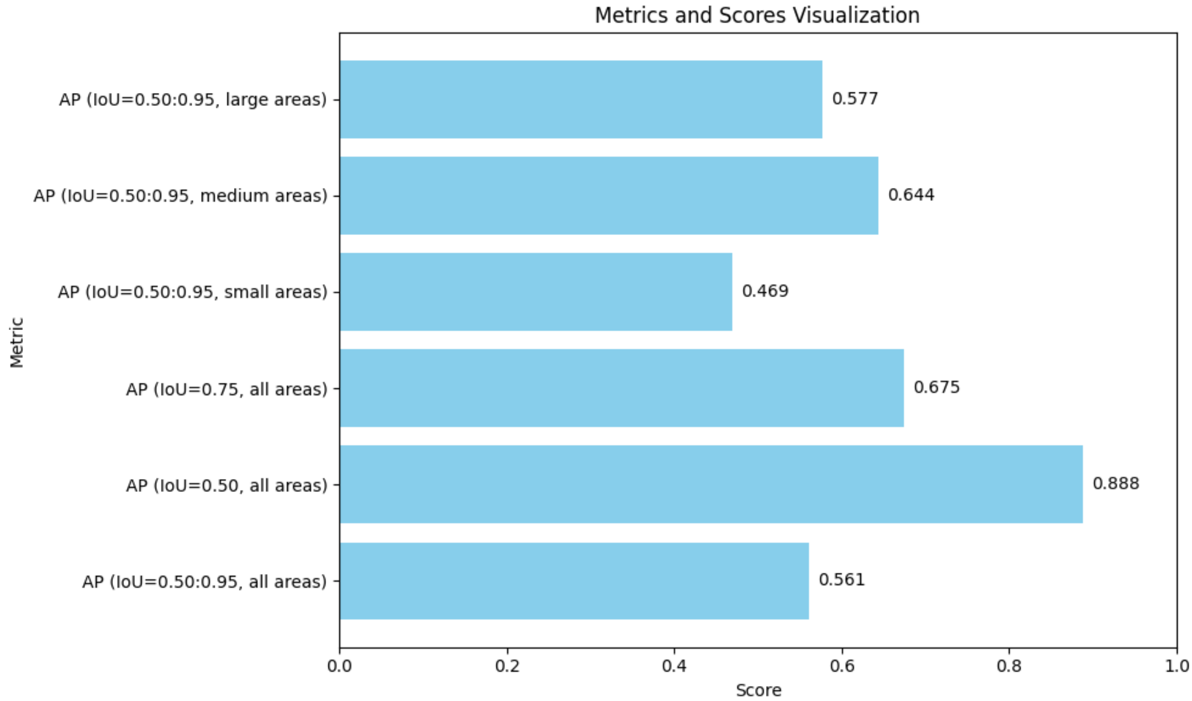
2. Initial Evaluation

2.1 Setup

The initial evaluation was conducted using a pre-trained DINO checkpoint downloaded from the official repository. This evaluation aimed to establish a baseline performance on our custom validation dataset.

2.2 Results

The following Average Precision (AP) scores were obtained on the validation dataset:

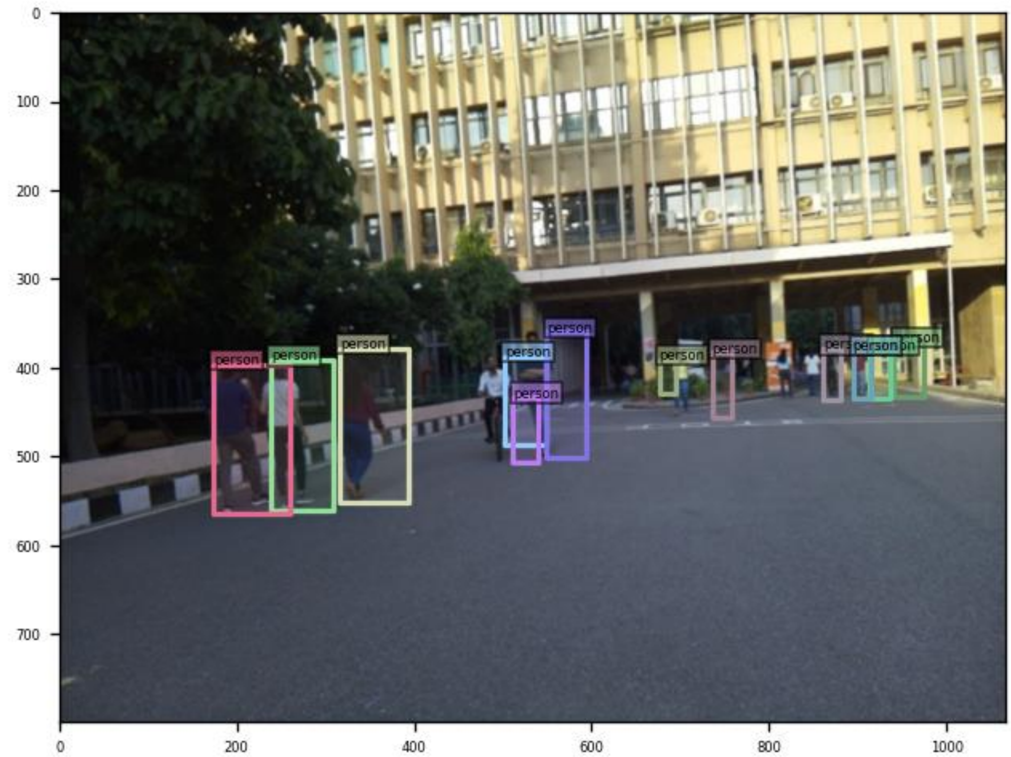


2.3 Observations and Analysis

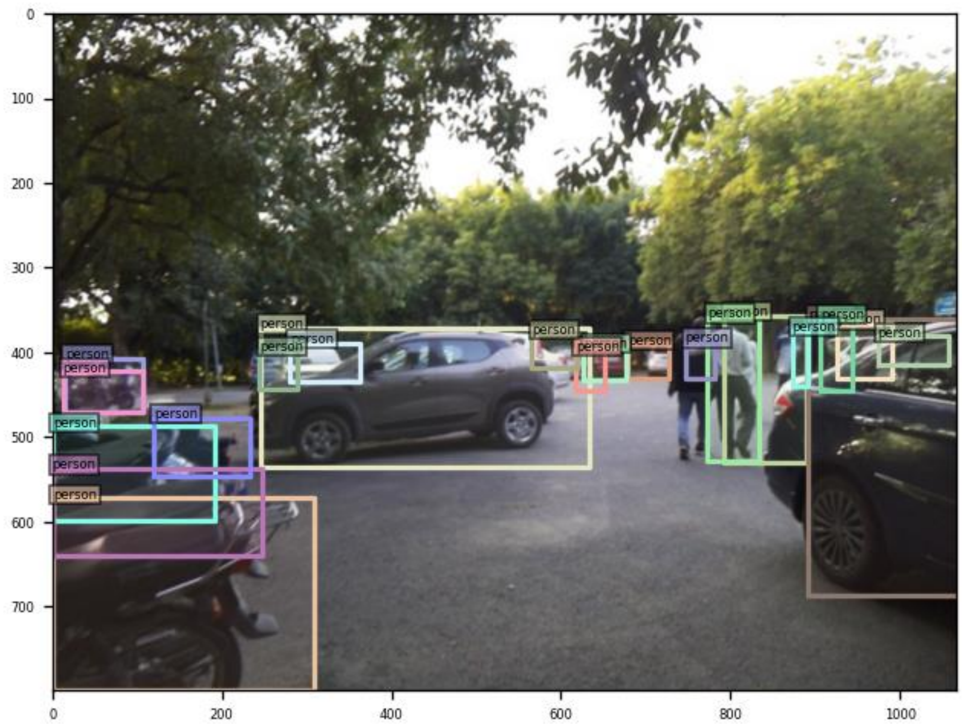
Upon examining the predictions, Two issues were identified:

1. **Bounding Box Misalignment:** In crowded scenes, the model correctly identified people but placed the bounding boxes a few centimeters away from where they were supposed to be.

This misalignment issue suggests that while the model is detecting the presence of people accurately, it's struggling with precise localization**.**



2. **Misclassification of Objects**: The model sometimes misidentified cars and bikes as persons. This suggests that the model may not have fully learned to distinguish between different object classes, particularly when they share similar shapes or contexts (e.g., vehicles often appear in similar environments as pedestrians).



These observations highlight areas for improvement in the model's ability to handle scale variations, crowded scenes, and distinguish between different object classes in urban environments.

3. Fine-Tuning Experiments

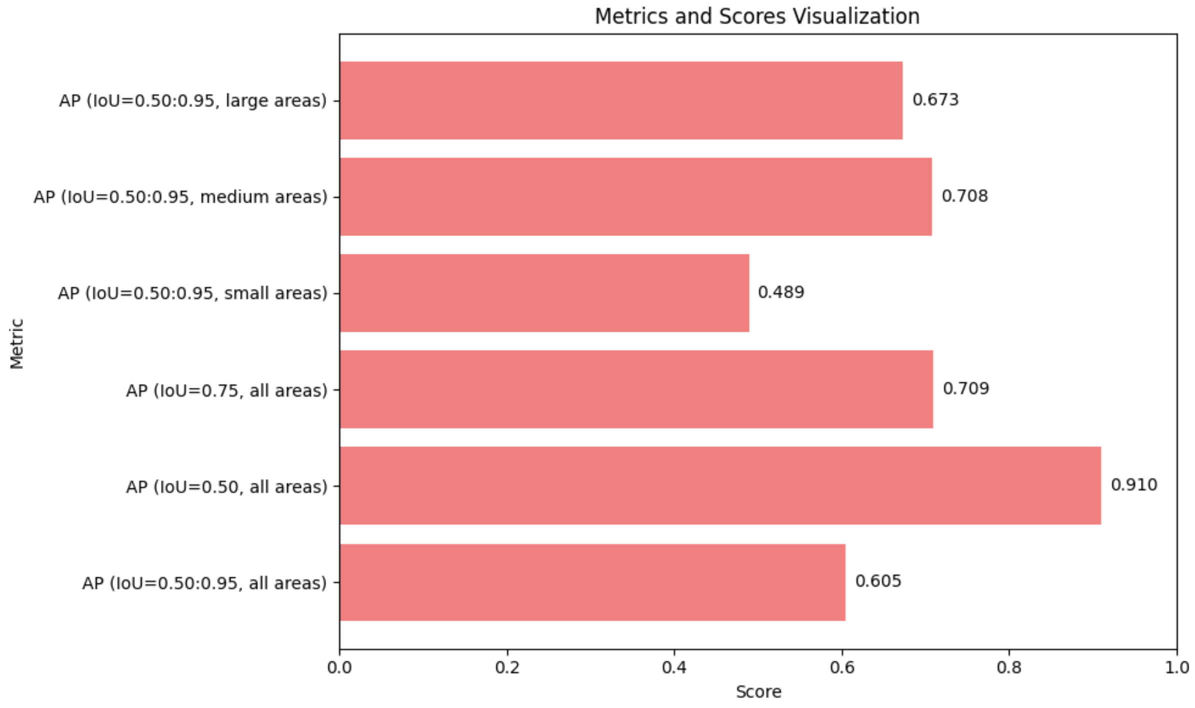
To address the identified issues and improve performance, two fine-tuning experiments were conducted by modifying the DINO configuration.

3.1 Experiment 1

Settings:

- Number of classes: 2
- Learning rate: 0.0001
- Epochs: 12
- Batch size: 2
- Dropout: 0.0

Results:



Analysis:

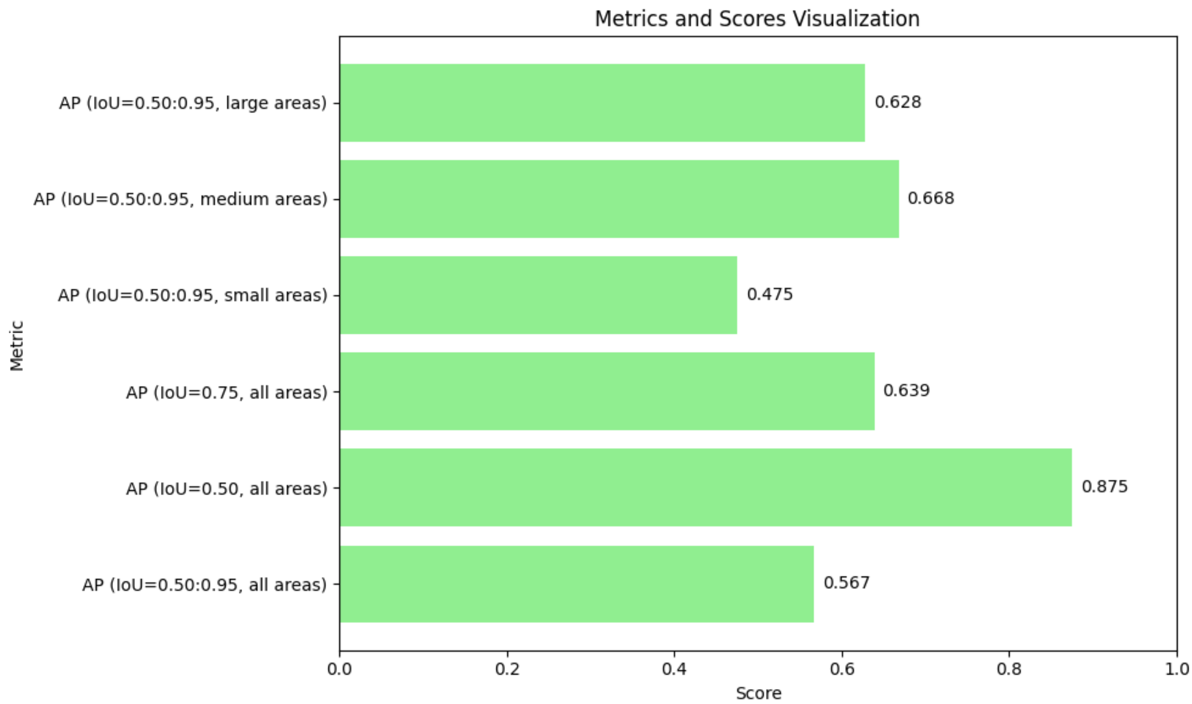
Experiment 1 showed a significant increase in accuracy compared to the initial evaluation. The model demonstrated improved performance across all metrics, with notable enhancements in detecting objects in medium and large areas. This improvement suggests that fine-tuning helped the model adapt better to the specific characteristics of our custom dataset.

3.2 Experiment 2

Settings:

- Number of classes: 2
- Learning rate: 0.0001
- Epochs: 10
- Batch size: 2
- Dropout: 0.1

Results:



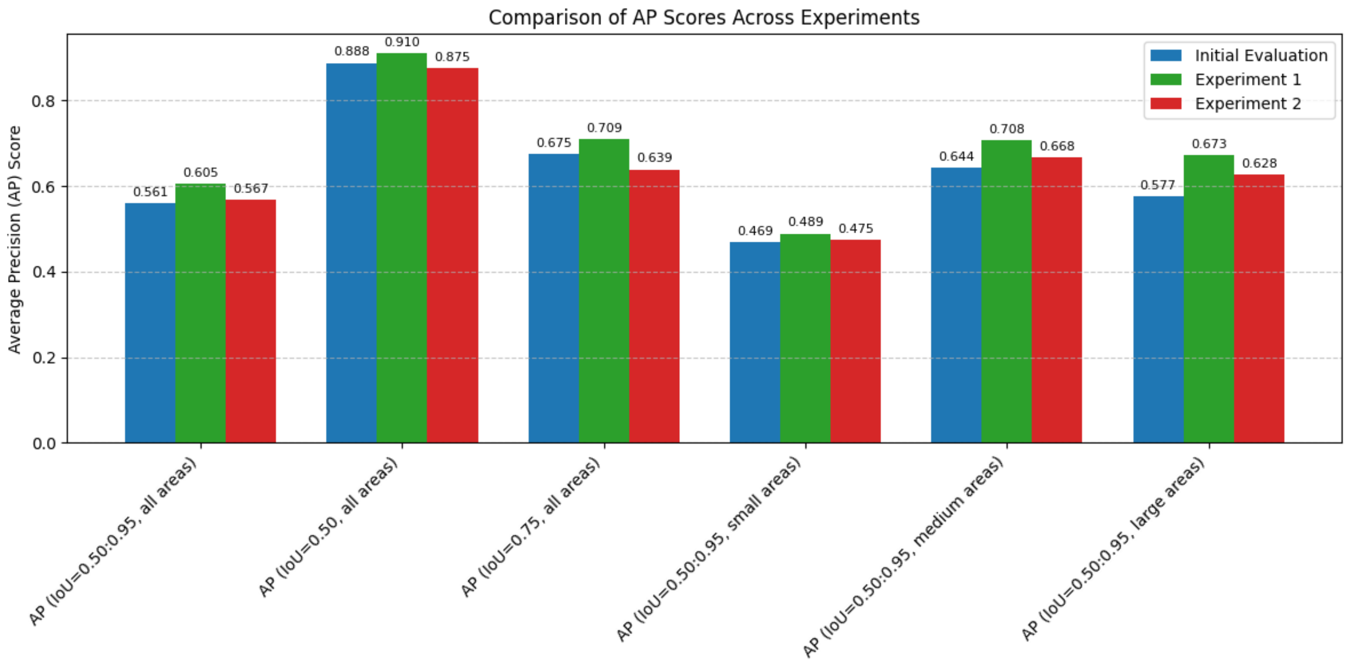
Analysis:

Experiment 2 showed lower performance compared to Experiment 1. The primary difference between the two experiments was the introduction of dropout (0.1) and a reduction in the number of epochs (10 instead of 12). The lower performance can be attributed to these changes:

1. **Dropout**: While dropout is generally used to prevent overfitting, in this case, it may have hindered the model's ability to learn task-specific features effectively. The dataset or the task complexity might not have required this level of regularization.
2. **Fewer Epochs**: The reduction in training epochs likely contributed to the lower performance. With only 10 epochs, the model may not have had sufficient time to fully adapt to the nuances of the custom dataset.

4. Conclusion and Results

Based on the experiments conducted, we can draw the following conclusions:



1. **Fine-tuning Efficacy**: Fine-tuning significantly improved the model's performance on our custom pedestrian detection dataset, addressing many of the issues observed in the initial evaluation.
2. **Optimal Configuration**: Experiment 1, with no dropout and 12 epochs of training, yielded the best results. This configuration allowed the model to learn effectively from the custom dataset without overfitting.
3. **Dropout Consideration**: For this specific task and dataset, dropout did not improve performance. This suggests that the model benefits more from learning task-specific features than from the regularization provided by dropout.
4. **Training Duration**: The longer training period (12 epochs) in Experiment 1 proved beneficial, allowing the model to better adapt to the dataset's characteristics.

Post-Finetuning Evaluation

After applying fine-tuning, the two issues - Bounding Box Misalignment and Misclassification of Objects , encountered during the initial evaluation were successfully resolved. The improvements in the Average Precision (AP) metrics, especially for smaller and medium areas, indicate that the fine-tuning process has addressed the inconsistencies observed earlier.

For more results check the ipynb file. Below are some results

Finetuned Exp 1 Results
Eval Results