Ali GÜVEN

STROMA: Machine Learning Challenge

Deep Learning Model Training: YOLO v5

Even if there is no official presentation of YOLO v5 like other family members (YOLO v2, v3, v7, v8), there are amount of literature studies about the architecture. It is very fast to deploy, easy to export other platforms like mobile, IoT. Besides, it has many architectural options to optimize for the usage, there are several level of backbones. Moreover, YOLO v5 was selected as the model for the problem. The model was obtained from [1].

As hyper-parameters, the default values were used like learning rate is 0.01, weight decay is 0.0005, momentum of Stochastic Gradient Descent optimization is 0.937. Training took about 10 hours on NVIDIA GTX 1050Ti GPU. No pre-trained model was used because in some of the real life problems, can be different from public dataset. This can cause bias. However, generally, pre-trained models can perform well. The parameters of training process are: Batch Size is 32, Number of epochs is 150, Image Size is 224, used model architecture is YOLOv5 medium. In Figure ,Figure 1 dataset instance distribution can be seen. There is an imbalance between classes (bolt and nut) to detect.
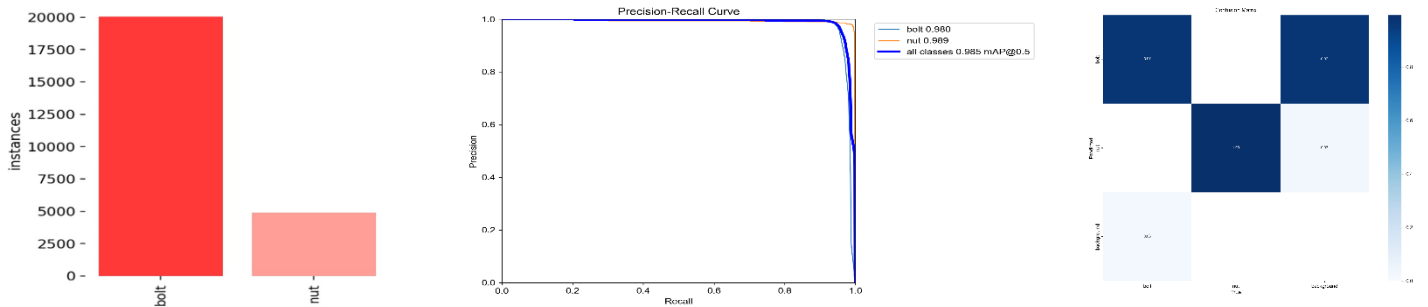
Figure 1: Instance Distribution (left) and Precision-Recall Curve on validation (middle), confusion matrix (right)
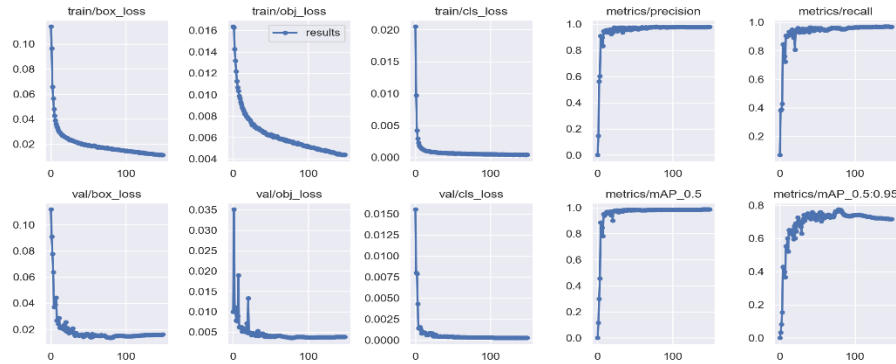
Figure 2: Training results, MAP50 is 0.985, MAP50-95 is 0.775 for all classes

The script to run training:

```
python ./yolov5/train.py --batch 32 --epochs 150 --data cv_data.yaml --weights '' --cfg
.yolov5/models/yolov5m.yaml --img 224 --workers 2
```

References

[1] https://ultralytics.com/yolov5