YOLO V5

```
#!python -m pip install --upgrade pip

#!pip install tensorflow==2.3.1

#!pip install tensorboard==2.4.1
!pip install torch
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: torch in /usr/local/lib/python3.8/dist-packages (1.13.0+cu116)
Requirement already satisfied: typing-extensions in /usr/local/lib/python3.8/dist-packages (from torch) (4.4.0)
```

```
import torch # YOLOv5 implemented using pytorch


from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
from IPython.display import Image #this is to render predictions


!git clone https://github.com/ultralytics/yolov5
```

```
Cloning into 'yolov5'...
remote: Enumerating objects: 14944, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (8/8), done.
remote: Total 14944 (delta 2), reused 3 (delta 0), pack-reused 14936
Receiving objects: 100% (14944/14944), 13.94 MiB | 29.42 MiB/s, done.
Resolving deltas: 100% (10266/10266), done.
```

```
%cd /content/yolov5
```

```
/content/yolov5
```

```
!pip install -r requirements.txt
```

```
  Downloading thop-0.1.1.post2209072238-py3-none-any.whl (15 kB)
Requirement already satisfied: torch>=1.7.0 in /usr/local/lib/python3.8/dist-packages (from -r requirements.txt (line 16)) (1.13.0+cu
Requirement already satisfied: torchvision>=0.8.1 in /usr/local/lib/python3.8/dist-packages (from -r requirements.txt (line 17)) (0.1
Requirement already satisfied: tqdm>=4.64.0 in /usr/local/lib/python3.8/dist-packages (from -r requirements.txt (line 18)) (4.64.1)
Requirement already satisfied: tensorboard>=2.4.1 in /usr/local/lib/python3.8/dist-packages (from -r requirements.txt (line 22)) (2.9
Requirement already satisfied: pandas>=1.1.4 in /usr/local/lib/python3.8/dist-packages (from -r requirements.txt (line 27)) (1.3.5)
Requirement already satisfied: seaborn>=0.11.0 in /usr/local/lib/python3.8/dist-packages (from -r requirements.txt (line 28)) (0.11.2
Collecting gitdb<5,>=4.0.1
  Downloading gitdb-4.0.10-py3-none-any.whl (62 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 62.7/62.7 KB 8.4 MB/s eta 0:00:00
Collecting jedi>=0.10
  Downloading jedi-0.18.2-py2.py3-none-any.whl (1.6 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.6/1.6 MB 38.0 MB/s eta 0:00:00
Requirement already satisfied: traitlets>=4.2 in /usr/local/lib/python3.8/dist-packages (from ipython->-r requirements.txt (line 6))
Requirement already satisfied: pygments in /usr/local/lib/python3.8/dist-packages (from ipython->-r requirements.txt (line 6)) (2.6.1
Requirement already satisfied: prompt-toolkit<2.1.0,>=2.0.0 in /usr/local/lib/python3.8/dist-packages (from ipython->-r requirements.
Requirement already satisfied: pexpect in /usr/local/lib/python3.8/dist-packages (from ipython->-r requirements.txt (line 6)) (4.8.0)
Requirement already satisfied: backcall in /usr/local/lib/python3.8/dist-packages (from ipython->-r requirements.txt (line 6)) (0.2.0
Requirement already satisfied: decorator in /usr/local/lib/python3.8/dist-packages (from ipython->-r requirements.txt (line 6)) (4.4.
Requirement already satisfied: pickleshare in /usr/local/lib/python3.8/dist-packages (from ipython->-r requirements.txt (line 6)) (0.
Requirement already satisfied: setuptools>=18.5 in /usr/local/lib/python3.8/dist-packages (from ipython->-r requirements.txt (line 6)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.8/dist-packages (from matplotlib>=3.2.2->-r requirements.txt (l
```

```
Requirement already satisfied: tensorboard-data-server<0.7.0,>=0.6.0 in /usr/local/lib/python3.8/dist-packages (from tensorboard>=2.4
Requirement already satisfied: google-auth<3,>=1.6.3 in /usr/local/lib/python3.8/dist-packages (from tensorboard>=2.4.1->-r requireme
Requirement already satisfied: wheel>=0.26 in /usr/local/lib/python3.8/dist-packages (from tensorboard>=2.4.1->-r requirements.txt (l
Requirement already satisfied: grpcio>=1.24.3 in /usr/local/lib/python3.8/dist-packages (from tensorboard>=2.4.1->-r requirements.txt
Requirement already satisfied: pytz>=2017.3 in /usr/local/lib/python3.8/dist-packages (from pandas>=1.1.4->-r requirements.txt (line
Collecting smmap<6,>=3.0.1
  Downloading smmap-5.0.0-py3-none-any.whl (24 kB)
Requirement already satisfied: cachetools<6.0,>=2.0.0 in /usr/local/lib/python3.8/dist-packages (from google-auth<3,>=1.6.3->tensorbo
Requirement already satisfied: six>=1.9.0 in /usr/local/lib/python3.8/dist-packages (from google-auth<3,>=1.6.3->tensorboard>=2.4.1->
Requirement already satisfied: pyasn1-modules>=0.2.1 in /usr/local/lib/python3.8/dist-packages (from google-auth<3,>=1.6.3->tensorboa
Requirement already satisfied: rsa<5,>=3.1.4 in /usr/local/lib/python3.8/dist-packages (from google-auth<3,>=1.6.3->tensorboard>=2.4.
Requirement already satisfied: requests-oauthlib>=0.7.0 in /usr/local/lib/python3.8/dist-packages (from google-auth-oauthlib<0.5,>=0.
Requirement already satisfied: parso<0.9.0,>=0.8.0 in /usr/local/lib/python3.8/dist-packages (from jedi>=0.10->ipython->-r requiremen
Requirement already satisfied: importlib-metadata>=4.4 in /usr/local/lib/python3.8/dist-packages (from markdown>=2.6.8->tensorboard>=
Requirement already satisfied: wcwidth in /usr/local/lib/python3.8/dist-packages (from prompt-toolkit<2.1.0,>=2.0.0->ipython->-r requ
Requirement already satisfied: ptyprocess>=0.5 in /usr/local/lib/python3.8/dist-packages (from pexpect->ipython->-r requirements.txt
Requirement already satisfied: zipp>=0.5 in /usr/local/lib/python3.8/dist-packages (from importlib-metadata>=4.4->markdown>=2.6.8->te
Requirement already satisfied: pyasn1<0.5.0,>=0.4.6 in /usr/local/lib/python3.8/dist-packages (from pyasn1-modules>=0.2.1->google-aut
Requirement already satisfied: oauthlib>=3.0.0 in /usr/local/lib/python3.8/dist-packages (from requests-oauthlib>=0.7.0->google-auth-
Installing collected packages: smmap, jedi, thop, gitdb, gitpython
```

Divide the dataset in train and val folder.

```
%cd /content/yolov5
```

```
/content/yolov5
```

Create dataset.yaml

move the dataset's dataset.yaml file to the /content/yolov5/data directory.

This file contains information required by YOLO to train the model on the custom data.

```
!python train.py --img 415 --batch 32 --epochs 30 --data /content/yolov5/data/coco128.yaml  --weights yolov5l.pt --cache
```

```
Epoch    GPU_mem   box_loss   obj_loss   cls_loss  Instances       Size
28/29      10.3G    0.03782    0.03752   0.003845        367        416: 100% 49/49 [00:26<00:00,  1.83it/s]
           Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 7/7 [00:04<00:00,  1.54it/s]
             all        448       3442      0.784      0.631      0.679      0.398

Epoch    GPU_mem   box_loss   obj_loss   cls_loss  Instances       Size
29/29      10.3G    0.03693    0.03561   0.003666        362        416: 100% 49/49 [00:29<00:00,  1.68it/s]
           Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 7/7 [00:04<00:00,  1.54it/s]
             all        448       3442      0.766      0.638      0.677      0.398

chs completed in 0.284 hours.
zer stripped from runs/train/exp/weights/last.pt, 92.7MB
zer stripped from runs/train/exp/weights/best.pt, 92.7MB

ting runs/train/exp/weights/best.pt...
 layers...
summary: 267 layers, 46113663 parameters, 0 gradients, 107.7 GFLOPs
           Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 7/7 [00:08<00:00,  1.19s/it]
             all        448       3442      0.783      0.631      0.679      0.398
          person        448       2179      0.763      0.588      0.639      0.322
         vehicle        448       1263      0.803      0.674      0.718      0.473
```

At the end of the training, two files should be saved in yolov5/runs/train/exp/weights: last.pt and best.pt. We'll use best.pt.

Explore the metrics recorded during training, I suggest you use TensorBoard, a very interactive exploration tool:

Let's explore now how confident our model is. We can plot a validation batch obtained during training and inspect the confidence score of each label

```
Image(filename= '/content/drive/MyDrive/happy_monk/New folder/happy monk/data/valid/images/image_000001240_jpg.rf.9617dc81c6e26819e3a9b3a5a67
```

Training losses and performance metrics are also logged to Tensorboard and a custom results.txt logfile which is plotted as results.png (below) after training completes.

You'll be implementing the detect.py script with the best.pt weights and image dimensions of 416x416 pixels (it's really important to comply with that). The results will be saved to runs/detect/exp. To display the results, run the following code:

```
# lets find the bounding boxes
```

```
#!python detect.py --source runs/train/exp/testimg.jpg --weights runs/train/exp/weights/best.pt --conf 0.25

!python detect.py --source '/content/drive/MyDrive/happy_monk/New folder/happy monk/data/valid/images/image_000001240_jpg.
    ve/MyDrive/happy_monk/New folder/happy monk/data/valid/images/image_000001240_jpg.rf.9617dc81c6e26819e3a9b3a5a6732cd5.jpg, data=data/coc



    ge_000001240_jpg.rf.9617dc81c6e26819e3a9b3a5a6732cd5.jpg: 640x640 3 persons, 4 vehicles, 51.1ms
    0)
```

```
# Model has detected the image " 3 persons, 4 vehicles,"


Image(filename= '/content/yolov5/runs/detect/exp5/image_000001240_jpg.rf.9617dc81c6e26819e3a9b3a5a6732cd5.jpg', width=1000)
```


```
Image(filename= "/content/drive/MyDrive/happy_monk/New folder/happy monk/data/valid/images/image_000000101_jpg.rf.1a5a509e0b7da7b2b8b43ad306a
```

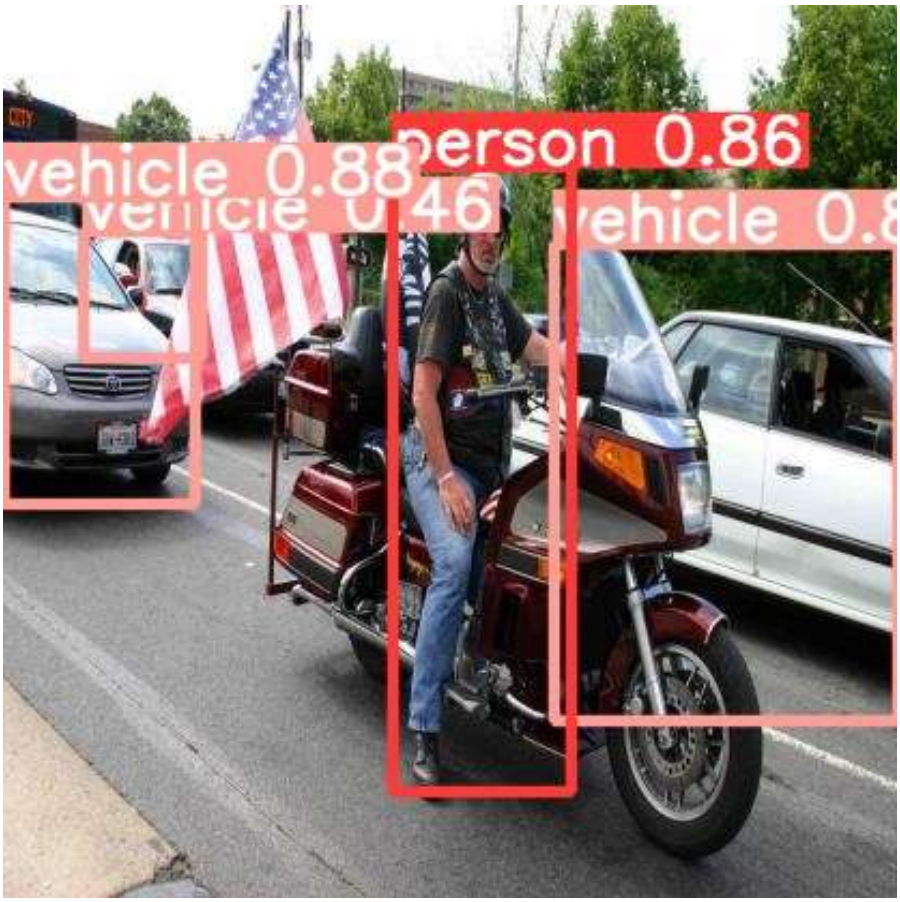
```
#!python detect.py --source runs/train/exp/testimg.jpg --weights runs/train/exp/weights/best.pt --conf 0.25

!python detect.py --source '/content/drive/MyDrive/happy_monk/New folder/happy monk/data/valid/images/image_000000101_jpg.rf.1a5a509e0b7da7b2
```

```
detect: weights=['/content/yolov5/runs/train/exp/weights/best.pt'], source=/content/drive/MyDrive/happy_monk/New folder/happy monk/data
YOLOv5 🚀 v7.0-61-gcaba2ae Python-3.8.16 torch-1.13.0+cu116 CUDA:0 (Tesla T4, 15110MiB)

Fusing layers...
Model summary: 267 layers, 46113663 parameters, 0 gradients, 107.7 GFLOPs
image 1/1 /content/drive/MyDrive/happy_monk/New folder/happy monk/data/valid/images/image_000000101_jpg.rf.1a5a509e0b7da7b2b8b43ad306ab
Speed: 0.6ms pre-process, 52.5ms inference, 1.5ms NMS per image at shape (1, 3, 640, 640)
Results saved to runs/detect/exp4
```

```
(filename= '/content/yolov5/runs/detect/exp4/image_000000101_jpg.rf.1a5a509e0b7da7b2b8b43ad306abf834.jpg', width=800)
```



✓ 0s completed at 12:26 PM