**How to run YOLOv11 on NVIDIA DGX**

* **Log in to Kubeflow** at [https://192.168.12.1:31380/](https://10.21.20.4:31380/) using your credentials on the FIREFOX web browser
* **Start a new notebook server**
  + PyTorch Image
  + 8 CPU, 16 Ram
  + 1 NVIDIA 10GB or 20 GB GPU (as per instructions)
  + No persistent storage
* **Start a new terminal in Kubeflow, then execute (all subsequent commands need to be executed in the terminal)**
  + *cd /workspace/*
* **Install Ultralytics**
  + *pip install ultralytics*
* **Install the necessary packages** 
  + *apt update*
  + *apt install ffmpeg libsm6 libxext6 -y*
* **To test the inference capabilities of Yolov11, run the command**
  + **Upload the test image and video**
  + **task = detect, segment, classify**
  + **mode = train , val, predict**
  + y*olo task=detect mode=predict model=yolo11n.pt source=”testimg.jpg”*
  + **AttributeError(might come)**: module 'numpy' has no attribute 'object'.

np.object` was a deprecated alias for the builtin `object`. To avoid this error in existing code, use `object` by itself. Doing this will not modify any behavior and is safe.

The aliases was originally deprecated in NumPy 1.20; for more details and guidance see the original release note at:

[https://numpy.org/devdocs/release/1.20.0-notes.html#deprecations](https://numpy.org/devdocs/release/1.20.0-notes.html" \l "deprecations)

***Solution: pip install numpy==1.23.4***

* + **[OPTIONAL]** By default, YOLOv11 tries to run on GPU. If you want to run it specifically on CPU, use the flag “--device cpu” as follows:
    - *pyolo task=detect mode=predict model=yolo11n.pt source=”testimg.jpg” device=’cpu’*



Fig: testimg.jpg

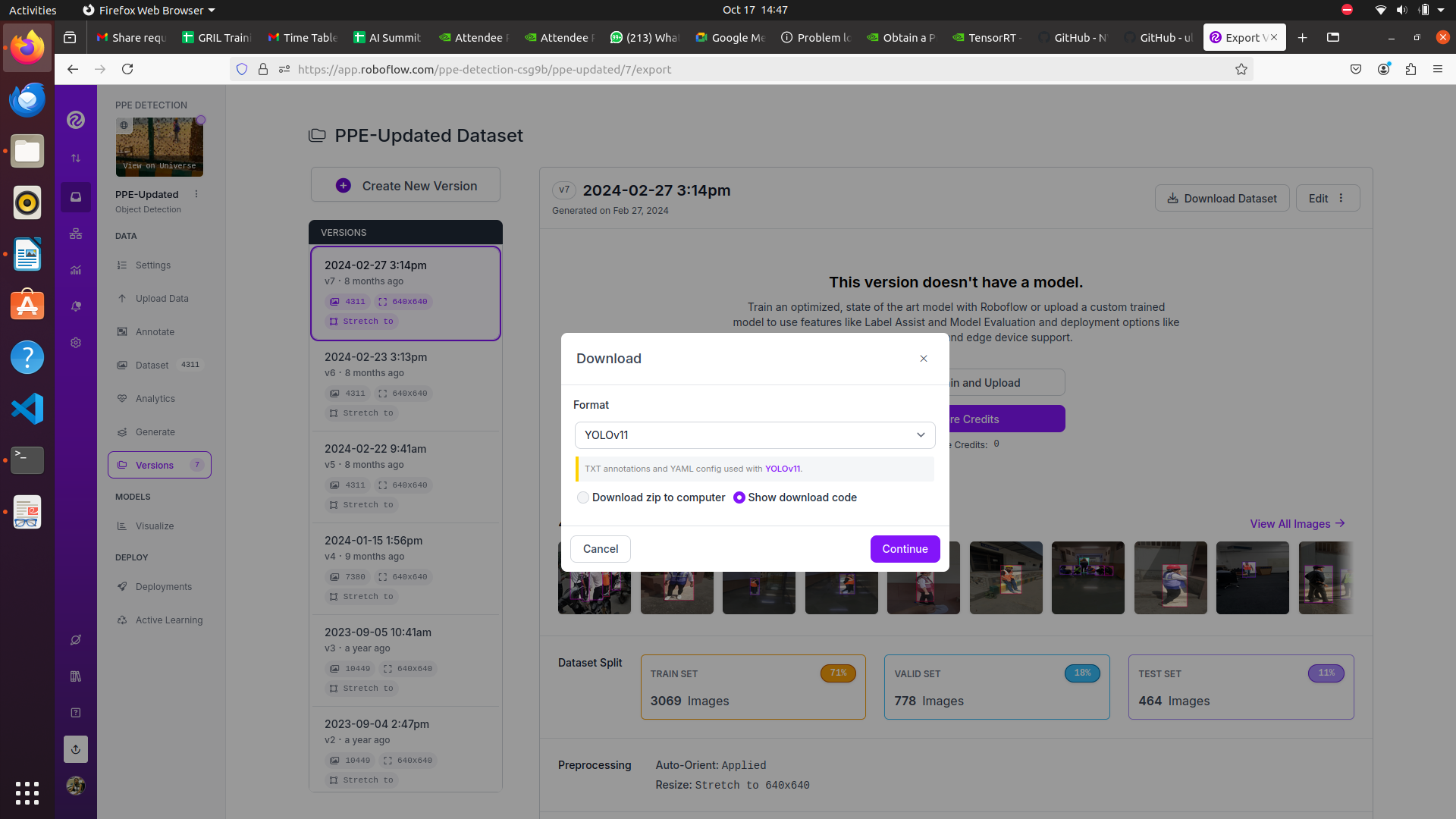


Fig: runs/detect/predict/testimg.jpg

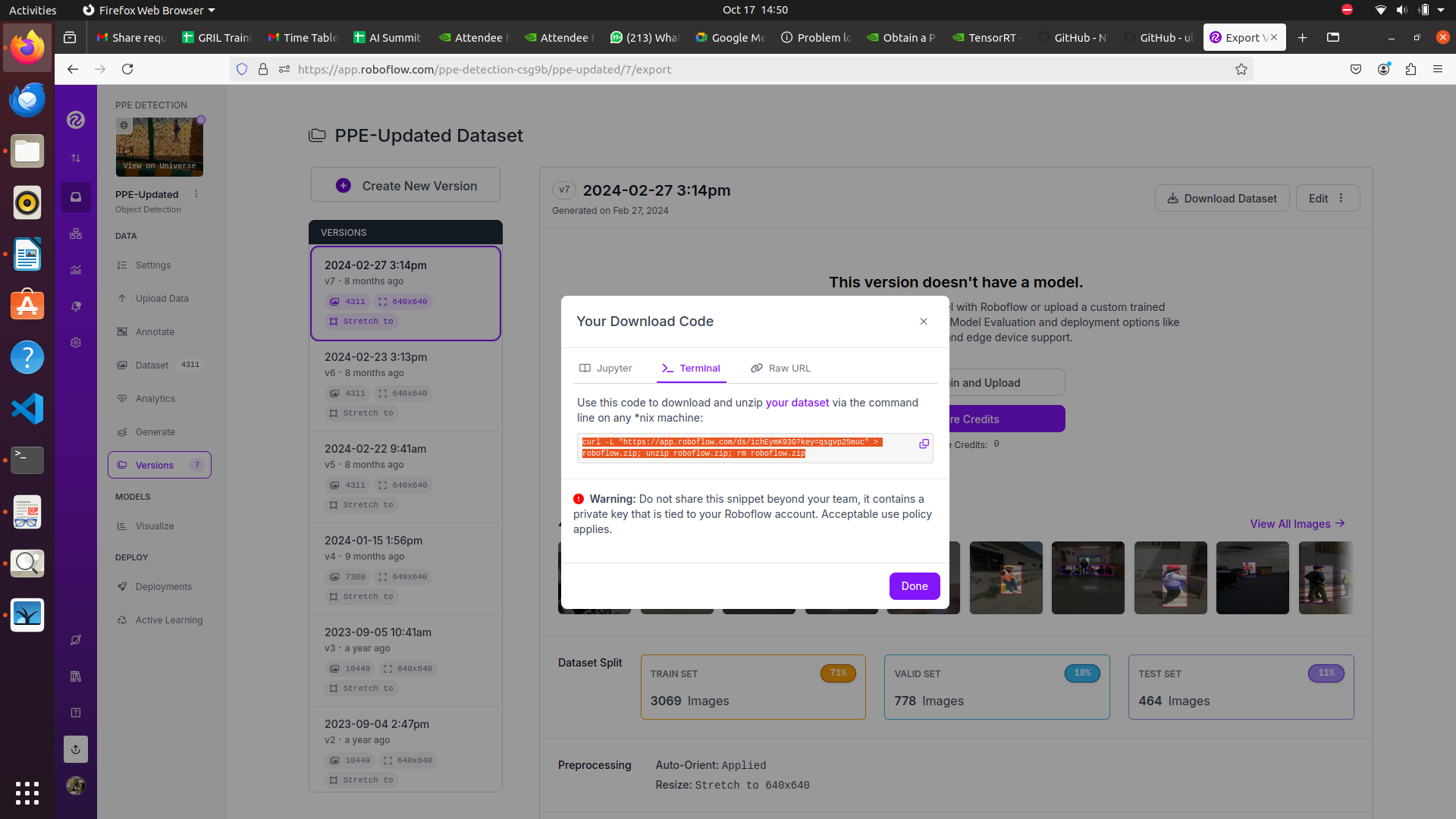
**How to train yolov11 on custom dataset using NVIDIA DGX**

***Note*** *- all the code needs to be executed in the Terminal in yolov9/ directory*

* **Export annotated dataset from Roboflow** 
  + Goto Roboflow -> Generate Dataset
  + In the preprocessing section, select the *Resize* parameter to 640\*640 then select *Generate*
  + Goto *Version* then select the dataset and select *Export*
  + The format is *YOLOv11*
  + Select *show download code* ->



* + Copy the code under the *Terminal* tab



* + **Note**: Temporarily, you can use the below code for downloading the annotated dataset from my account into DGX

*curl -L "https://app.roboflow.com/ds/jxJaNTK8Ff?key=kdmFLi71DX" > roboflow.zip; unzip roboflow.zip; rm roboflow.zip*

* **Download the annotated dataset into DGX -** Paste the download code in the Terminal of the Kubeflow notebook

*mkdir custom\_dataset*

*cd custom\_dataset/*

*curl -L "https://app.roboflow.com/ds/jxJaNTK8Ff?key=kdmFLi71DX" > roboflow.zip*

*unzip roboflow.zip; rm roboflow.zip*

*cd ..*

* **Update Custom Dataset configuration**
  + Open *data.yaml* file by browsing to c*ustom\_dataset/* directory in Kubeflow File Browser
  + Update train, val, and test fields as given below:

*train: /workspace/custom\_dataset/train/images*

*val: /workspace/custom\_dataset/valid/images*

*test: /workspace/custom\_dataset/test/images*

* + **Note**: Do not make any other changes to the file
  + Rename the file *data.yaml* to *custom\_data.yaml*

* **Start the training process**

* *yolo task=detect mode=train model=yolo11n.pt data="custom\_dataset/custom\_data.yaml" epochs=10 imgsz=640*
* ***U might observe:*** *the training speed is very slow, which means that your code is not using GPU, instead it is using CPU*

***Solution:*** *You have to match the version compatibility of torch and CUDA*

***For this first of all check the cuda version by:***

*nvcc –version or*

*print(torch.version.cuda) (at ABESIT it is 11.1 currently, might change later)*

* *pip uninstall torch torchvision torchaudio***(first uninstall torch)**
* **Then install torch as per cuda version again**

*pip install torch==1.10.0+cu111 torchvision==0.11.0+cu111 torchaudio==0.10.0 -f* [*https://download.pytorch.org/whl/torch\_stable.html*](https://download.pytorch.org/whl/torch_stable.html) *(***u can get this link from pytorch versions on google as per your cuda version)**

* **Run the training command (above) again**
* We can use *--device cpu* with the command if we are training the model on CPU
* **Explore the results of training**
  + Results saved to ***runs/detect/train***directory
  + The directory ***runs/detect/train/weights*** contains the weights of the model
    - The file *best.pt* contains the best weights of the trained model

**TASKS**

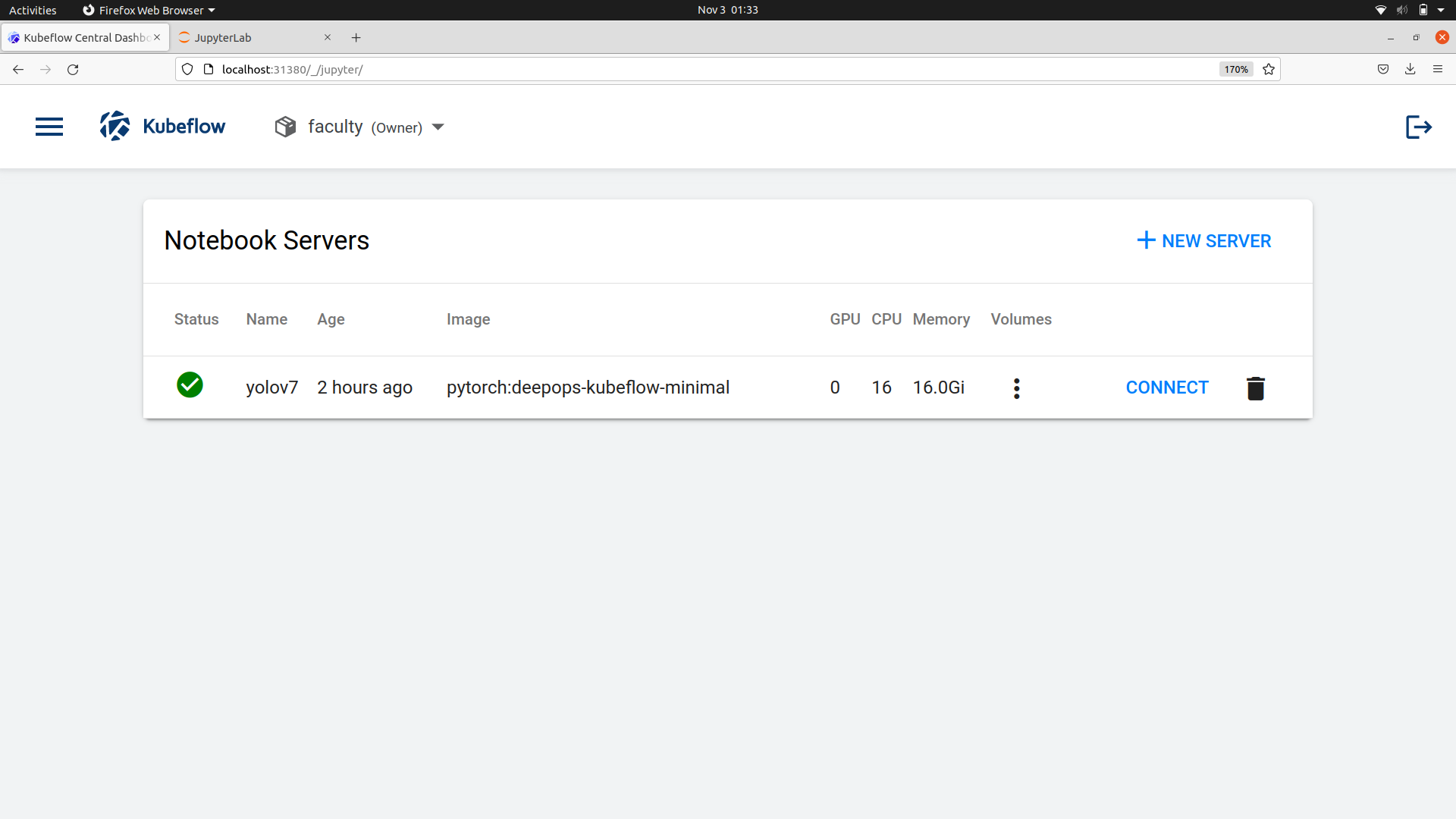
* **Do inference on custom data**
  + **Copy the trained model** (best.pt) from

***runs/detect/train/weights to /workspace***

* + - *cp runs/detect/train/weights/best.pt /workspace*
  + **Test model on your custom data**
    - Upload test image or video or both
    - Upload test images to /workspacedirectory using Kubeflow’s file browser
    - Run the following script in Terminal
    - *yolo task=detect mode=predict model=best.pt source=”vid.mp4”*
* View the results in *runs/detect/predict* directory

**VERY IMPORTANT:** You always need to delete your Kubeflow notebook as soon as you finish your task since it blocks GPU to be utilized by other students!!!

If you fail to do so then your account will be deactivated



**(Ref link: https://github.com/ultralytics/ultralytics.git)**