```
!pip install opendatasets
```

```
Requirement already satisfied: opendatasets in /usr/local/lib/python3.11/dist-packages (0.1.22)
Requirement already satisfied: tqdm in /usr/local/lib/python3.11/dist-packages (from opendatasets) (
Requirement already satisfied: kaggle in /usr/local/lib/python3.11/dist-packages (from opendatasets)
Requirement already satisfied: click in /usr/local/lib/python3.11/dist-packages (from opendatasets)
Requirement already satisfied: bleach in /usr/local/lib/python3.11/dist-packages (from kaggle->opend
Requirement already satisfied: certifi>=14.05.14 in /usr/local/lib/python3.11/dist-packages (from ka
Requirement already satisfied: charset-normalizer in /usr/local/lib/python3.11/dist-packages (from k
Requirement already satisfied: idna in /usr/local/lib/python3.11/dist-packages (from kaggle->opendat
Requirement already satisfied: protobuf in /usr/local/lib/python3.11/dist-packages (from kaggle->ope
Requirement already satisfied: python-dateutil>=2.5.3 in /usr/local/lib/python3.11/dist-packages (fr
Requirement already satisfied: python-slugify in /usr/local/lib/python3.11/dist-packages (from kaggl
Requirement already satisfied: requests in /usr/local/lib/python3.11/dist-packages (from kaggle->ope
Requirement already satisfied: setuptools>=21.0.0 in /usr/local/lib/python3.11/dist-packages (from k
Requirement already satisfied: six>=1.10 in /usr/local/lib/python3.11/dist-packages (from kaggle->op
Requirement already satisfied: text-unidecode in /usr/local/lib/python3.11/dist-packages (from kaggl
Requirement already satisfied: urllib3>=1.15.1 in /usr/local/lib/python3.11/dist-packages (from kagg
Requirement already satisfied: webencodings in /usr/local/lib/python3.11/dist-packages (from kaggle-
```

```
import opendatasets as od
od.download("https://www.kaggle.com/datasets/mustafatayyipbayram/ppe-detection")
```

```
Skipping, found downloaded files in "./ppe-detection" (use force=True to force download)
```

```
!pip install ultralytics
```

```
Requirement already satisfied: ultralytics in /usr/local/lib/python3.11/dist-packages (8.3.162)
Requirement already satisfied: numpy>=1.23.0 in /usr/local/lib/python3.11/dist-packages (from ultral
Requirement already satisfied: matplotlib>=3.3.0 in /usr/local/lib/python3.11/dist-packages (from ul
Requirement already satisfied: opencv-python>=4.6.0 in /usr/local/lib/python3.11/dist-packages (from
Requirement already satisfied: pillow>=7.1.2 in /usr/local/lib/python3.11/dist-packages (from ultral
Requirement already satisfied: pyyaml>=5.3.1 in /usr/local/lib/python3.11/dist-packages (from ultral
Requirement already satisfied: requests>=2.23.0 in /usr/local/lib/python3.11/dist-packages (from ult
Requirement already satisfied: scipy>=1.4.1 in /usr/local/lib/python3.11/dist-packages (from ultraly
Requirement already satisfied: torch>=1.8.0 in /usr/local/lib/python3.11/dist-packages (from ultraly
Requirement already satisfied: torchvision>=0.9.0 in /usr/local/lib/python3.11/dist-packages (from u
Requirement already satisfied: tqdm>=4.64.0 in /usr/local/lib/python3.11/dist-packages (from ultraly
Requirement already satisfied: psutil in /usr/local/lib/python3.11/dist-packages (from ultralytics)
Requirement already satisfied: py-cpuinfo in /usr/local/lib/python3.11/dist-packages (from ultralyti
Requirement already satisfied: pandas>=1.1.4 in /usr/local/lib/python3.11/dist-packages (from ultral
Requirement already satisfied: ultralytics-thop>=2.0.0 in /usr/local/lib/python3.11/dist-packages (f
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.11/dist-packages (from mat
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.11/dist-packages (from matplot
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.11/dist-packages (from ma
Requirement already satisfied: kiwisolver>=1.3.1 in /usr/local/lib/python3.11/dist-packages (from ma
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.11/dist-packages (from matp
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.11/dist-packages (from mat
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.11/dist-packages (from
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas>
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from panda
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.11/dist-packages (
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dist-packages (from request
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from r
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from r
Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-packages (from torch>=1.8.
Requirement already satisfied: typing-extensions>=4.10.0 in /usr/local/lib/python3.11/dist-packages
Requirement already satisfied: networkx in /usr/local/lib/python3.11/dist-packages (from torch>=1.8.
Requirement already satisfied: jinja2 in /usr/local/lib/python3.11/dist-packages (from torch>=1.8.0-
Requirement already satisfied: fsspec in /usr/local/lib/python3.11/dist-packages (from torch>=1.8.0-
Requirement already satisfied: sympy==1.13.1 in /usr/local/lib/python3.11/dist-packages (from torch>
Requirement already satisfied: mpmath<1.4,>=1.1.0 in /usr/local/lib/python3.11/dist-packages (from s
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from python-date
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.11/dist-packages (from jinj
```

```
from ultralytics import YOLO
```

```
import torch
torch.cuda.empty_cache()
```

```
import torch
from torch.utils.data import TensorDataset
```

```
import os
def load_yolo_labels(path1):
    labels = []
    with open(path1, 'r') as
        for line in f:
            parts = list(map(float, line.strip().split()))
```

005335_jpg.rf.f97b56c •••

What can I help you build?

```
                labels.append(parts)
        # Convert to FloatTensor (optional, you can just use the list)
        tensor = torch.FloatTensor(labels)
        return tensor

    def save_yolo_labels(path, tensor_labels):
        # tensor_labels is a torch.FloatTensor or list of lists
        with open(path, 'w') as f:
            for label in tensor_labels:
                line = ' '.join(str(x.item() if isinstance(x, torch.Tensor) else x) for x in label)
                f.write(line + '\n')

    label_dir = "/content/ppe-detection/data/train/labels"

    for file_name in os.listdir(label_dir):
        if file_name.endswith(".txt"):
            full_path = os.path.join(label_dir, file_name)

            # Load labels
            labels_tensor = load_yolo_labels(full_path)
            print(f"✅ Labels for {file_name}:")
            print(labels_tensor)

            # (Optional) Modify labels_tensor here if you want to fix or change labels

            # Save (overwrite) original label file with same labels (or modified)
            save_yolo_labels(full_path, labels_tensor)
            print(f"🔄 Overwritten file: {full_path}")
```

```
model = YOLO("yolov8n.pt")  # detection model, NOT segmentation
results = model.train(data="custom_data.yaml", epochs=10, batch=8, imgsz=640, task="detect")
```

```
Ultralytics 8.3.162 🚀 Python-3.11.13 torch-2.6.0+cpu CPU (Intel Xeon 2.00GHz)
engine/trainer: agnostic_nms=False, amp=True, augment=False, auto_augment=randaugment, batch=8, bg
Overriding model.yaml nc=80 with nc=2

                   from  n    params  module                                       arguments
  0                  -1  1       464  ultralytics.nn.modules.conv.Conv             [3, 16, 3, 2]
  1                  -1  1      4672  ultralytics.nn.modules.conv.Conv             [16, 32, 3, 2]
  2                  -1  1      7360  ultralytics.nn.modules.block.C2f             [32, 32, 1, Tru
  3                  -1  1     18560  ultralytics.nn.modules.conv.Conv             [32, 64, 3, 2]
  4                  -1  2     49664  ultralytics.nn.modules.block.C2f             [64, 64, 2, Tru
  5                  -1  1     73984  ultralytics.nn.modules.conv.Conv             [64, 128, 3, 2]
  6                  -1  2    197632  ultralytics.nn.modules.block.C2f             [128, 128, 2, T
  7                  -1  1    295424  ultralytics.nn.modules.conv.Conv             [128, 256, 3, 2
  8                  -1  1    460288  ultralytics.nn.modules.block.C2f             [256, 256, 1, T
  9                  -1  1    164608  ultralytics.nn.modules.block.SPPF            [256, 256, 5]
 10                  -1  1         0  torch.nn.modules.upsampling.Upsample         [None, 2, 'near
 11             [-1, 6]  1         0  ultralytics.nn.modules.conv.Concat           [1]
 12                  -1  1    148224  ultralytics.nn.modules.block.C2f             [384, 128, 1]
 13                  -1  1         0  torch.nn.modules.upsampling.Upsample         [None, 2, 'near
 14             [-1, 4]  1         0  ultralytics.nn.modules.conv.Concat           [1]
 15                  -1  1     37248  ultralytics.nn.modules.block.C2f             [192, 64, 1]
 16                  -1  1     36992  ultralytics.nn.modules.conv.Conv             [64, 64, 3, 2]
 17            [-1, 12]  1         0  ultralytics.nn.modules.conv.Concat           [1]
 18                  -1  1    123648  ultralytics.nn.modules.block.C2f             [192, 128, 1]
 19                  -1  1    147712  ultralytics.nn.modules.conv.Conv             [128, 128, 3, 2
 20             [-1, 9]  1         0  ultralytics.nn.modules.conv.Concat           [1]
 21                  -1  1    493056  ultralytics.nn.modules.block.C2f             [384, 256, 1]
 22        [15, 18, 21]  1    751702  ultralytics.nn.modules.head.Detect           [2, [64, 128, 2
Model summary: 129 layers, 3,011,238 parameters, 3,011,222 gradients, 8.2 GFLOPs

Transferred 319/355 items from pretrained weights
Freezing layer 'model.22.dfl.conv.weight'
train: Fast image access ✅ (ping: 0.0±0.0 ms, read: 772.6±203.9 MB/s, size: 28.1 KB)
train: Scanning /content/ppe-detection/data/test/labels.cache... 1766 images, 0 backgrounds, 0 cor

val: Scanning /content/ppe-detection/data/val/labels.cache... 1581 images, 0 backgrounds, 0 corrup

optimizer: 'optimizer=auto' found, ignoring 'lr0=0.01' and 'momentum=0.937' and determining best '
optimizer: AdamW(lr=0.001667, momentum=0.9) with parameter groups 57 weight(decay=0.0), 64 weight(
Image sizes 640 train, 640 val
Using 0 dataloader workers
Logging results to runs/detect/train15
Starting training for 10 epochs...
Closing dataloader mosaic

      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
       1/10         0G      1.529      1.927      1.232         34        640: 100%|██████████| 22
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100%|███
                   all       1581       7823      0.803       0.68      0.766       0.43

      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
       2/10         0G       1.45      1.315      1.202         37        640: 100%|██████████| 22
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100%|███
                   all       1581       7823      0.819      0.718        0.8      0.458

      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
       3/10         0G      1.433      1.133      1.186         33        640: 100%|██████████| 22
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100%|███
```

```
predits = model.predict(source="/content/ppe-detection/data/test/images/005335_jpg.rf.f97b56c8e44
```

```
image 1/1 /content/ppe-detection/data/test/images/005335_jpg.rf.f97b56c8e4497a72479dfd9426596a7c.jpg
Speed: 3.0ms preprocess, 50.9ms inference, 1.0ms postprocess per image at shape (1, 3, 640, 640)
Results saved to runs/detect/train154
```

```
import matplotlib.pyplot as plt

for predict in predits:
    predict.plot()
    plt.imshow(predict.plot())
    plt.axis('off')
    plt.show()
```


```
for predit in predits:
    boxes = predit.boxes
    print(boxes.xyxy)
    print(boxes.conf)
    print(boxes.cls)
```

```
tensor([[181.7989, 191.5065, 195.0265, 206.5802]])
tensor([0.7834])
tensor([1.])
```

Start coding or generate with AI.