

Model training conversion

Model training conversion

1. Model training
 2. Model conversion
- References

After completing the tutorial content of dataset annotation, we can use the motherboard to start training the model.

This tutorial only introduces the model training and conversion of CLI. You can refer to the official website to modify the Python case

1. Model training

Use CLI command to train the model directly: copy the yolo11n.pt file to the directory where the configuration file is located, and then open the terminal in the directory where the configuration file is located:

```
cd /home/jetson/ultralytics/ultralytics/data/yahboom_data/orange_data
```

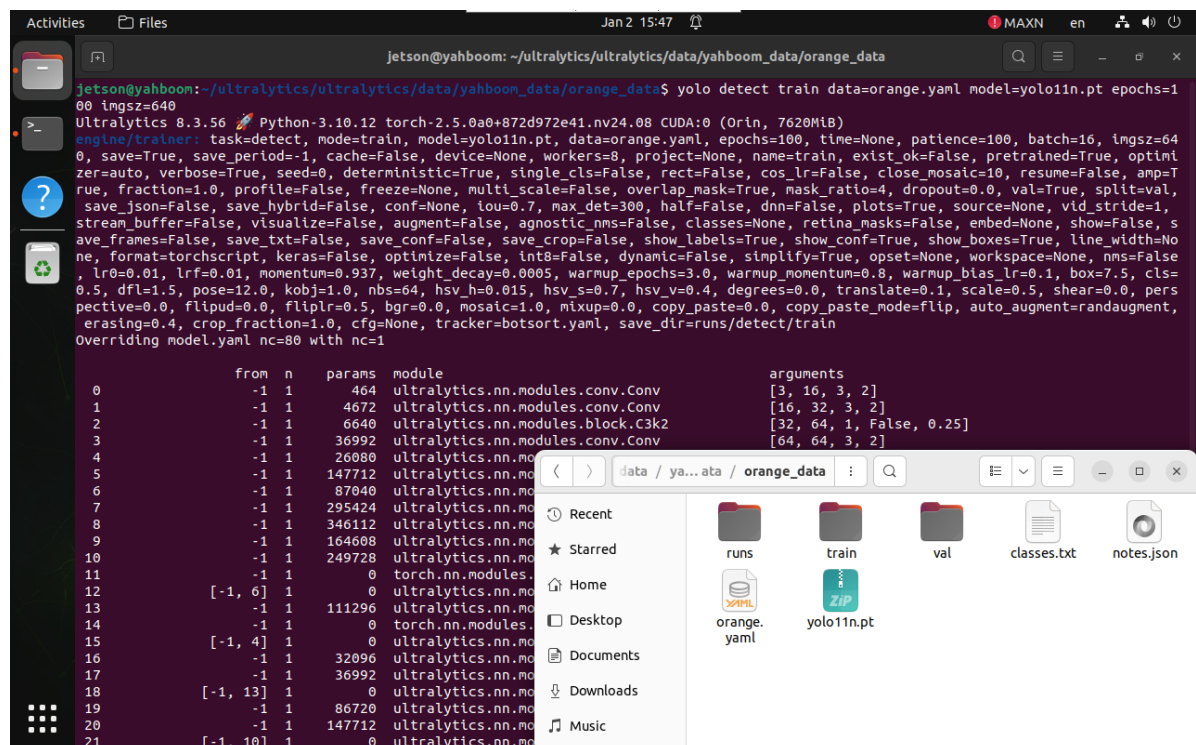
```
yolo detect train data=orange.yaml model=yolo11n.pt epochs=100 imgsz=640
```

data: Dataset configuration file

model: Pre-trained model file

epochs: Number of training rounds

imgsz: Enter the specified image size



```

Activities Terminal Jan 2 15:47 MAXN en
System throttled due to Over-current.

17      -1 1 36992 u
18      [-1, 13] 1 0 u
19      -1 1 86720 ultralytics.nn.modules.block.C3k2 [192, 128, 1, False]
20      -1 1 147712 ultralytics.nn.modules.conv.Conv [128, 128, 3, 2]
21      [-1, 10] 1 0 ultralytics.nn.modules.conv.Concat [1]
22      -1 1 378880 ultralytics.nn.modules.block.C3k2 [384, 256, 1, True]
23      [16, 19, 22] 1 430867 ultralytics.nn.modules.head.Detect [1, [64, 128, 256]]
YOLO11n summary: 319 layers, 2,590,035 parameters, 2,590,019 gradients, 6.4 GFLOPs

Transferred 448/499 items from pretrained weights
Freezing layer 'model.23.dfl.conv.weight'
AMP: running Automatic Mixed Precision (AMP) checks...
AMP: checks passed
train: Scanning /home/jetson/ultralytics/ultralytics/data/yahboom_data/orange_data/train/labels... 153 images, 0 backgrounds, 0 corr
train: New cache created: /home/jetson/ultralytics/ultralytics/data/yahboom_data/orange_data/train/labels.cache
val: Scanning /home/jetson/ultralytics/ultralytics/data/yahboom_data/orange_data/val/labels... 20 images, 0 backgrounds, 0 corrupt:
val: New cache created: /home/jetson/ultralytics/ultralytics/data/yahboom_data/orange_data/val/labels.cache
Plotting labels to runs/detect/train/labels.jpg...
optimizer: 'optimizer-auto' found, ignoring 'lr0=0.01' and 'momentum=0.937' and determining best 'optimizer', 'lr0' and 'momentum' a
utomatically...
optimizer: AdamW(lr=0.002, momentum=0.9) with parameter groups 81 weight(decay=0.0), 88 weight(decay=0.0005), 87 bias(decay=0.0)
Image sizes 640 train, 640 val
Using 6 dataloader workers
Logging results to runs/detect/train
Starting training for 100 epochs...

Epoch GPU_mem box_loss cls_loss dfl_loss Instances Size
1/100 2.35G 0.5434 2.647 0.9827 19 640: 100%| 10/10 [00:08<00:00, 1.14it/s]
Class Images Instances Box(P R mAP50 mAP50-95): 100%| 1/1 [00:00<00:00, 2.03it
all 20 20 0.00333 1 0.995 0.888

Epoch GPU_mem box_loss cls_loss dfl_loss Instances Size
2/100 2.36G 0.5282 1.707 0.9395 21 640: 100%| 10/10 [00:05<00:00, 1.74it/s]
Class Images Instances Box(P R mAP50 mAP50-95): 100%| 1/1 [00:00<00:00, 3.21it
all 20 20 0.00333 1 0.995 0.879

Epoch GPU_mem box_loss cls_loss dfl_loss Instances Size
3/100 2.38G 0.5424 1.267 0.972 16 640: 100%| 10/10 [00:05<00:00, 1.71it/s]
Class Images Instances Box(P R mAP50 mAP50-95): 100%| 1/1 [00:00<00:00, 3.27it
all 20 20 0.00838 1 0.995 0.912

```

```

Activities Terminal Jan 2 16:03 MAXN en
jetson@yahboom: ~/ultralytics/ultralytics/data/yahboom_data/orange_data

all 20 20 0.997 1 0.995 0.943

Epoch GPU_mem box_loss cls_loss dfl_loss Instances Size
96/100 2.38G 0.2912 0.2331 0.8209 9 640: 100%| 10/10 [00:05<00:00, 1.96it/s]
Class Images Instances Box(P R mAP50 mAP50-95): 100%| 1/1 [00:00<00:00, 3.06it
all 20 20 0.997 1 0.995 0.954

Epoch GPU_mem box_loss cls_loss dfl_loss Instances Size
97/100 2.37G 0.2719 0.2317 0.8218 9 640: 100%| 10/10 [00:05<00:00, 1.98it/s]
Class Images Instances Box(P R mAP50 mAP50-95): 100%| 1/1 [00:00<00:00, 3.41it
all 20 20 0.997 1 0.995 0.946

Epoch GPU_mem box_loss cls_loss dfl_loss Instances Size
98/100 2.38G 0.2796 0.2191 0.8276 9 640: 100%| 10/10 [00:05<00:00, 1.96it/s]
Class Images Instances Box(P R mAP50 mAP50-95): 100%| 1/1 [00:00<00:00, 3.34it
all 20 20 0.997 1 0.995 0.962

Epoch GPU_mem box_loss cls_loss dfl_loss Instances Size
99/100 2.37G 0.2753 0.2198 0.8061 9 640: 100%| 10/10 [00:05<00:00, 1.99it/s]
Class Images Instances Box(P R mAP50 mAP50-95): 100%| 1/1 [00:00<00:00, 3.20it
all 20 20 0.997 1 0.995 0.948

Epoch GPU_mem box_loss cls_loss dfl_loss Instances Size
100/100 2.38G 0.2593 0.2153 0.8236 9 640: 100%| 10/10 [00:05<00:00, 1.96it/s]
Class Images Instances Box(P R mAP50 mAP50-95): 100%| 1/1 [00:00<00:00, 3.45it
all 20 20 0.997 1 0.995 0.955

100 epochs completed in 0.182 hours.
Optimizer stripped from runs/detect/train/weights/last.pt, 5.5MB
Optimizer stripped from runs/detect/train/weights/best.pt, 5.5MB

Validating runs/detect/train/weights/best.pt...
Ultralytics 8.3.56 Python-3.10.12 torch-2.5.0a0+872d972e41.nv24.08 CUDA:0 (Orin, 7620MiB)
YOLO11n summary (fused): 238 layers, 2,582,347 parameters, 0 gradients, 6.3 GFLOPs
Class Images Instances Box(P R mAP50 mAP50-95): 100%| 1/1 [00:00<00:00, 2.46it
all 20 20 0.997 1 0.995 0.962

Speed: 0.8ms preprocess, 9.8ms inference, 0.0ms loss, 1.8ms postprocess per image
Results saved to runs/detect/train
Learn more at https://docs.ultralytics.com/modes/train
jetson@yahboom: ~/ultralytics/ultralytics/data/yahboom_data/orange_data$

```

2. Model conversion

The final model will be generated in the runs folder: generally choose the best.pt file for use

```
/home/jetson/ultralytics/ultralytics/data/yahboom_data/orange_data/runs/detect/train/weights
```

Convert the PyTorch model to TensorRT:

```
cd
/home/jetson/ultralytics/ultralytics/data/yahboom_data/orange_data/runs/detect/train/weights
```

```
yolo export model=best.pt format=engine
```

```
Activities Terminal Jan 2 16:14 MAXN en
jetson@yahboom: ~/ultralytics/ultralytics/data/yahboom_data/orange_data/runs/detect/train/weights
jetson@yahboom:~/ultralytics/ultralytics/data/yahboom_data/orange_data/runs/detect/train/weights$ ls
best.pt last.pt
jetson@yahboom:~/ultralytics/ultralytics/data/yahboom_data/orange_data/runs/detect/train/weights$ yolo export model=best.pt format=engine
WARNING ⚠️TensorRT requires GPU export, automatically assigning device=0
Ultralytics 8.3.56 Python-3.10.12 torch-2.5.0a0+872d972e41.nv24.08 CUDA:0 (Orin, 7620MiB)
YOLO11n summary (fused): 238 layers, 2,582,347 parameters, 0 gradients, 6.3 GFLOPs

PyTorch: starting from 'best.pt' with input shape (1, 3, 640, 640) BCHW and output shape(s) (1, 5, 8400) (5.2 MB)

ONNX: starting export with onnx 1.17.0 opset 19...
ONNX: slimming with onnxslim 0.1.45...
ONNX: export success ✅ 2.5s, saved as 'best.onnx' (10.1 MB)

TensorRT: starting export with TensorRT 10.3.0...
[01/02/2025-16:11:04] [TRT] [I] [MemUsageChange] Init CUDA: CPU +2, GPU +0, now: CPU 663, GPU 3380 (MiB)
[01/02/2025-16:11:06] [TRT] [I] [MemUsageChange] Init builder kernel library: CPU +927, GPU +609, now: CPU 1633, GPU 4009 (MiB)
[01/02/2025-16:11:06] [TRT] [I] -----
[01/02/2025-16:11:06] [TRT] [I] Input filename: best.onnx
[01/02/2025-16:11:06] [TRT] [I] ONNX IR version: 0.0.9
[01/02/2025-16:11:06] [TRT] [I] Opset version: 19
[01/02/2025-16:11:06] [TRT] [I] Producer name: pytorch
[01/02/2025-16:11:06] [TRT] [I] Producer version: 2.5.0
[01/02/2025-16:11:06] [TRT] [I] Domain:
[01/02/2025-16:11:06] [TRT] [I] Model version: 0
[01/02/2025-16:11:06] [TRT] [I] Doc string:
[01/02/2025-16:11:06] [TRT] [I] -----
TensorRT: input "images" with shape(1, 3, 640, 640) DataType.FLOAT
TensorRT: output "output0" with shape(1, 5, 8400) DataType.FLOAT
TensorRT: building FP32 engine as best.engine
[01/02/2025-16:11:06] [TRT] [I] Local timing cache in use. Profiling results in this builder pass will not be stored.
[01/02/2025-16:14:04] [TRT] [I] Detected 1 inputs and 1 output network tensors.
[01/02/2025-16:14:06] [TRT] [I] Total Host Persistent Memory: 540272
[01/02/2025-16:14:06] [TRT] [I] Total Device Persistent Memory: 38912
[01/02/2025-16:14:06] [TRT] [I] Total Scratch Memory: 2764800
[01/02/2025-16:14:06] [TRT] [I] [BlockAssignment] Started assigning block shifts. This will take 225 steps to complete.
[01/02/2025-16:14:06] [TRT] [I] [BlockAssignment] Algorithm ShiftNTopDown took 22.5896ms to assign 11 blocks to 225 nodes requiring
18842624 bytes.
[01/02/2025-16:14:06] [TRT] [I] Total Activation Memory: 18841600
[01/02/2025-16:14:06] [TRT] [I] Total Weights Memory: 10456516
```

```
Activities Terminal Jan 2 16:14 MAXN en
jetson@yahboom: ~/ultralytics/ultralytics/data/yahboom_data/orange_data/runs/detect/train/weights
TensorRT: starting export with TensorRT 10.3.0...
[01/02/2025-16:11:04] [TRT] [I] [MemUsageChange] Init CUDA: CPU +2, GPU +0, now: CPU 663, GPU 3380 (MiB)
[01/02/2025-16:11:06] [TRT] [I] [MemUsageChange] Init builder kernel library: CPU +927, GPU +609, now: CPU 1633, GPU 4009 (MiB)
[01/02/2025-16:11:06] [TRT] [I] -----
[01/02/2025-16:11:06] [TRT] [I] Input filename: best.onnx
[01/02/2025-16:11:06] [TRT] [I] ONNX IR version: 0.0.9
[01/02/2025-16:11:06] [TRT] [I] Opset version: 19
[01/02/2025-16:11:06] [TRT] [I] Producer name: pytorch
[01/02/2025-16:11:06] [TRT] [I] Producer version: 2.5.0
[01/02/2025-16:11:06] [TRT] [I] Domain:
[01/02/2025-16:11:06] [TRT] [I] Model version: 0
[01/02/2025-16:11:06] [TRT] [I] Doc string:
[01/02/2025-16:11:06] [TRT] [I] -----
TensorRT: input "images" with shape(1, 3, 640, 640) DataType.FLOAT
TensorRT: output "output0" with shape(1, 5, 8400) DataType.FLOAT
TensorRT: building FP32 engine as best.engine
[01/02/2025-16:11:06] [TRT] [I] Local timing cache in use. Profiling results in this builder pass will not be stored.
[01/02/2025-16:14:04] [TRT] [I] Detected 1 inputs and 1 output network tensors.
[01/02/2025-16:14:06] [TRT] [I] Total Host Persistent Memory: 540272
[01/02/2025-16:14:06] [TRT] [I] Total Device Persistent Memory: 38912
[01/02/2025-16:14:06] [TRT] [I] Total Scratch Memory: 2764800
[01/02/2025-16:14:06] [TRT] [I] [BlockAssignment] Started assigning block shifts. This will take 225 steps to complete.
[01/02/2025-16:14:06] [TRT] [I] [BlockAssignment] Algorithm ShiftNTopDown took 22.5896ms to assign 11 blocks to 225 nodes requiring
18842624 bytes.
[01/02/2025-16:14:06] [TRT] [I] Total Activation Memory: 18841600
[01/02/2025-16:14:06] [TRT] [I] Total Weights Memory: 10456516
[01/02/2025-16:14:06] [TRT] [I] Engine generation completed in 179.878 seconds.
[01/02/2025-16:14:06] [TRT] [I] [MemUsageStats] Peak memory usage of TRT CPU/GPU memory allocators: CPU 1 MiB, GPU 132 MiB
[01/02/2025-16:14:06] [TRT] [I] [MemUsageStats] Peak memory usage during Engine building and serialization: CPU: 2598 MiB
TensorRT: export success ✅ 185.6s, saved as 'best.engine' (11.8 MB)

Export complete (186.9s)
Results saved to /home/jetson/ultralytics/ultralytics/data/yahboom_data/orange_data/runs/detect/train/weights
Predict: yolo predict task=detect model=best.engine imgsz=640
Validate: yolo val task=detect model=best.engine imgsz=640 data=orange.yaml
Visualize: https://netron.app
💡 Learn more at https://docs.ultralytics.com/modes/export
jetson@yahboom:~/ultralytics/ultralytics/data/yahboom_data/orange_data/runs/detect/train/weights$ ls
best.engine best.onnx best.pt last.pt
jetson@yahboom:~/ultralytics/ultralytics/data/yahboom_data/orange_data/runs/detect/train/weights$
```

References

<https://docs.ultralytics.com/modes/train/>