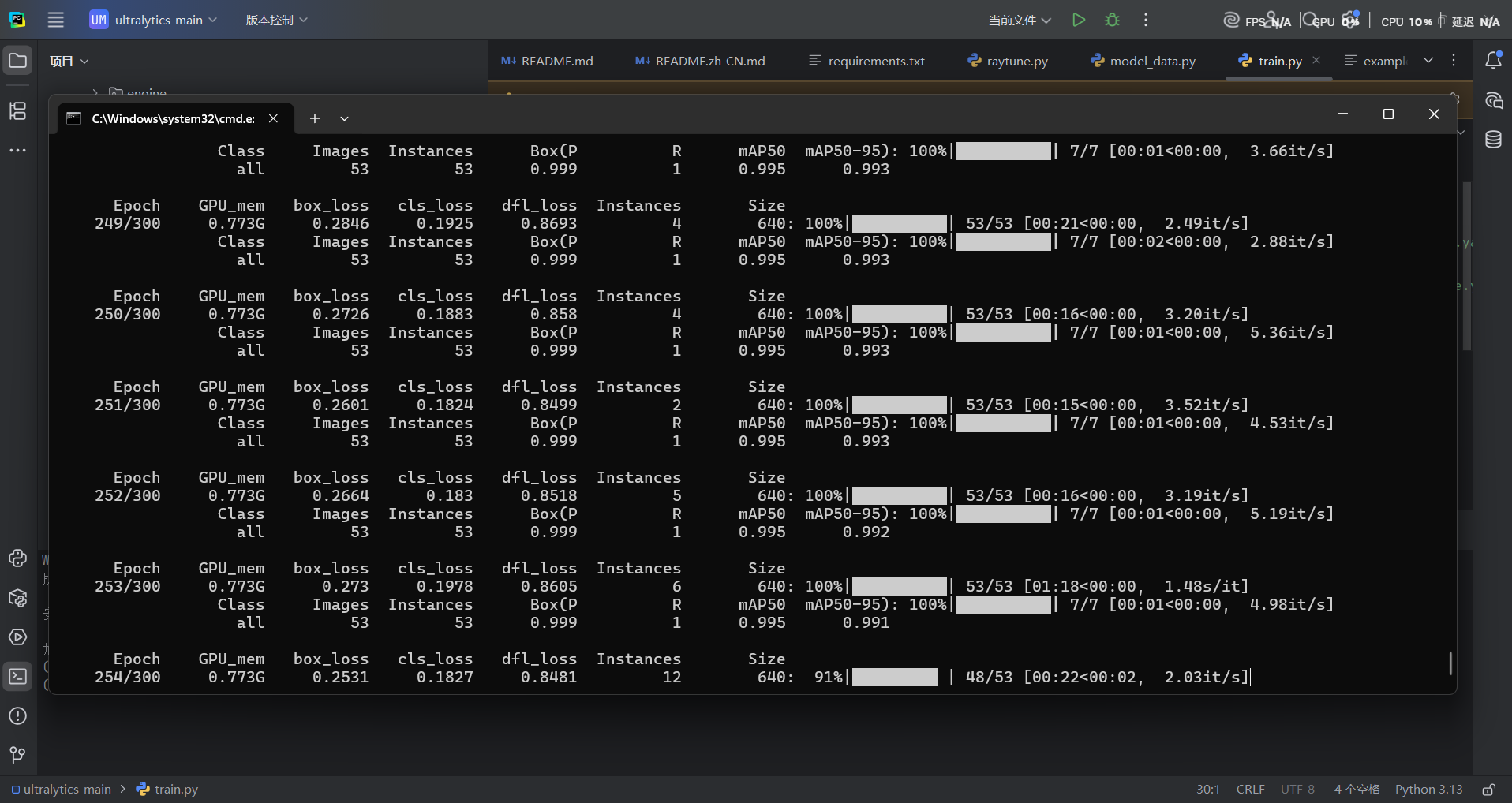
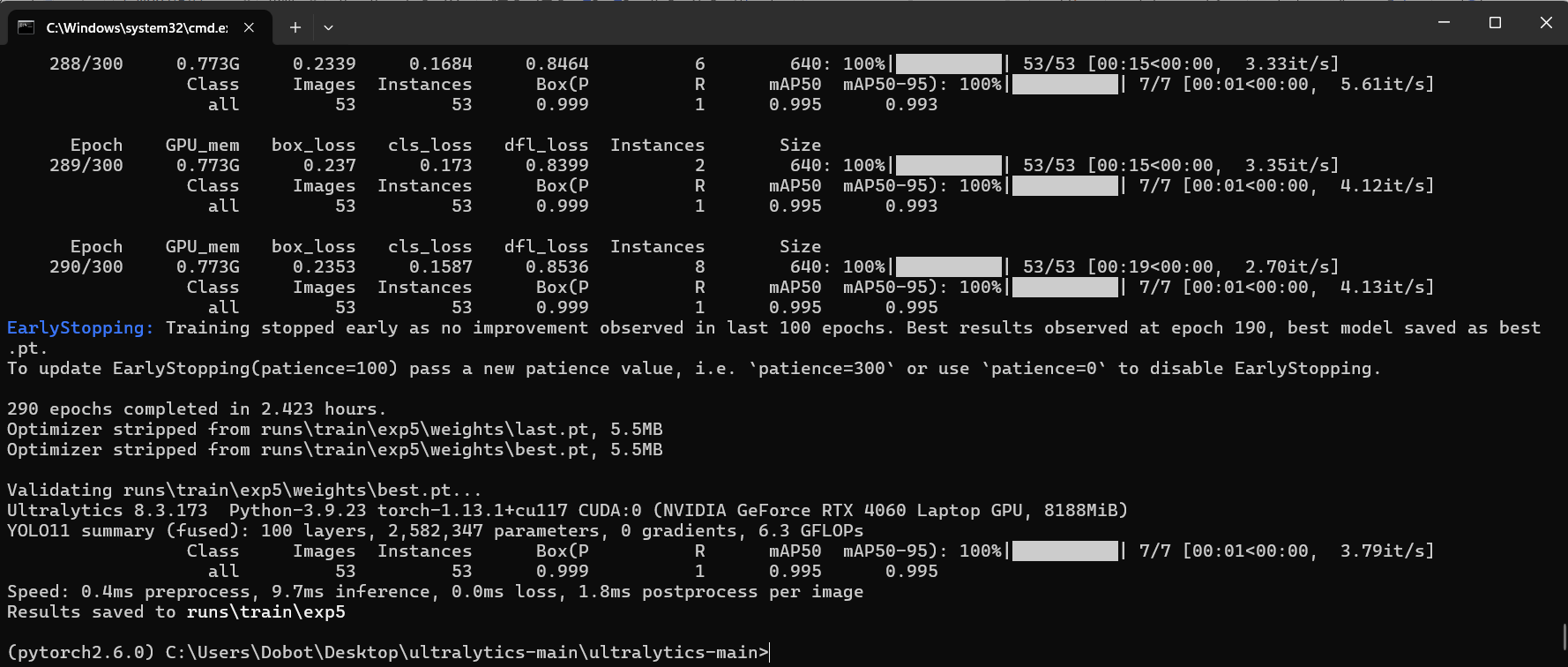
YOLOv11在tiger-pose数据集上进行训练：

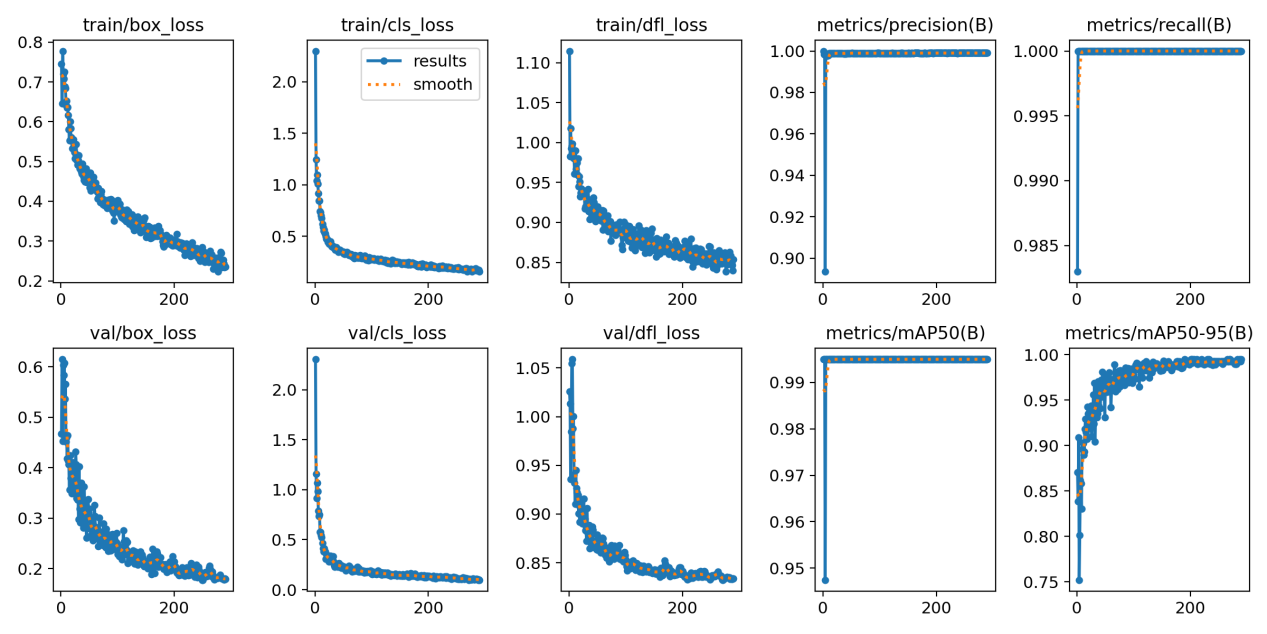




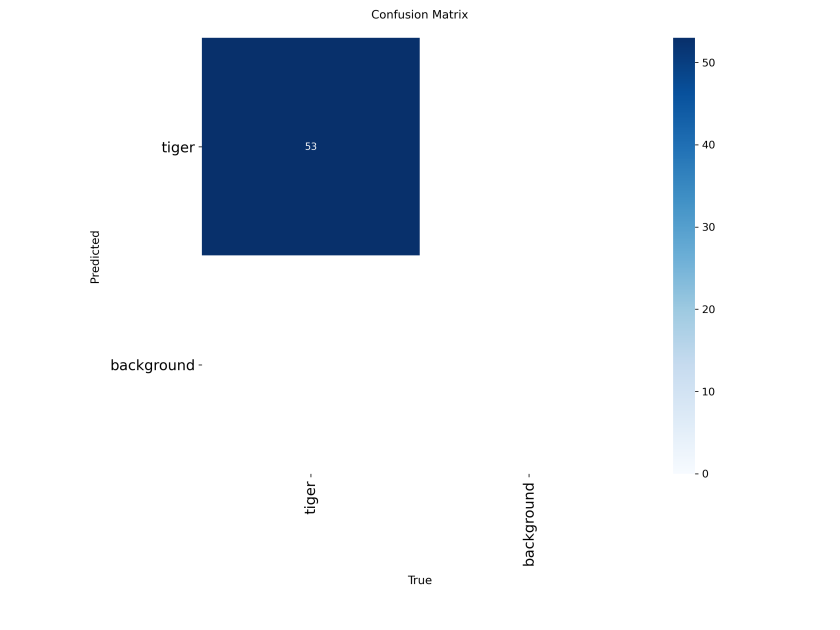
参考网页：

<https://blog.csdn.net/m0_53414416/article/details/147092841>

训练、验证过程的损失函数：



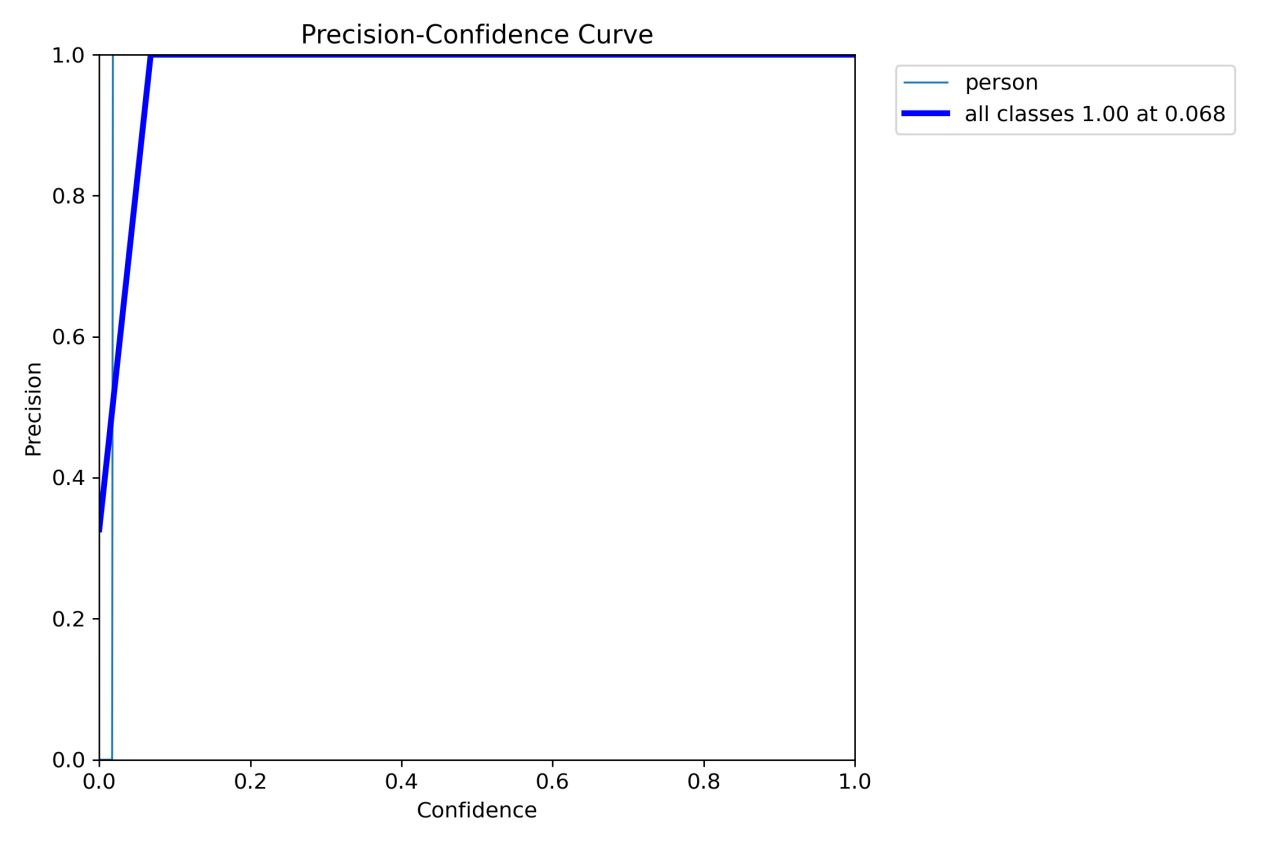
混淆矩阵：



验证结果可视化：



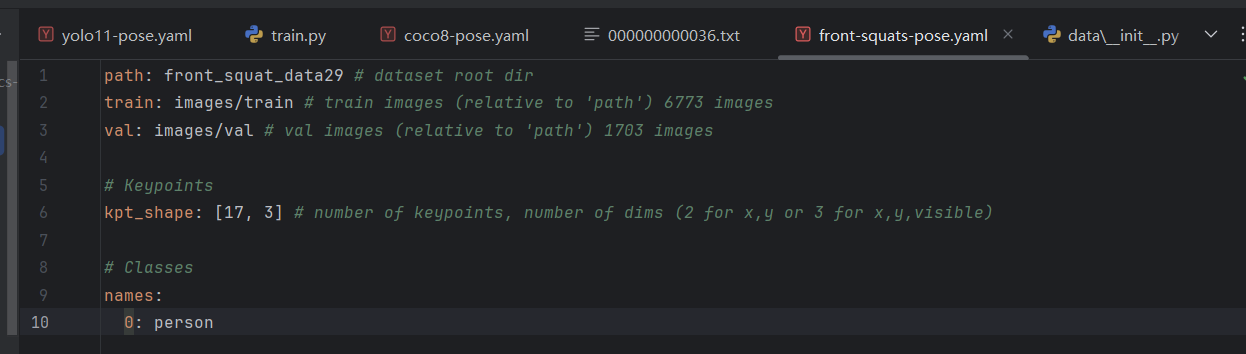
在coco8-pose数据集上进行训练：



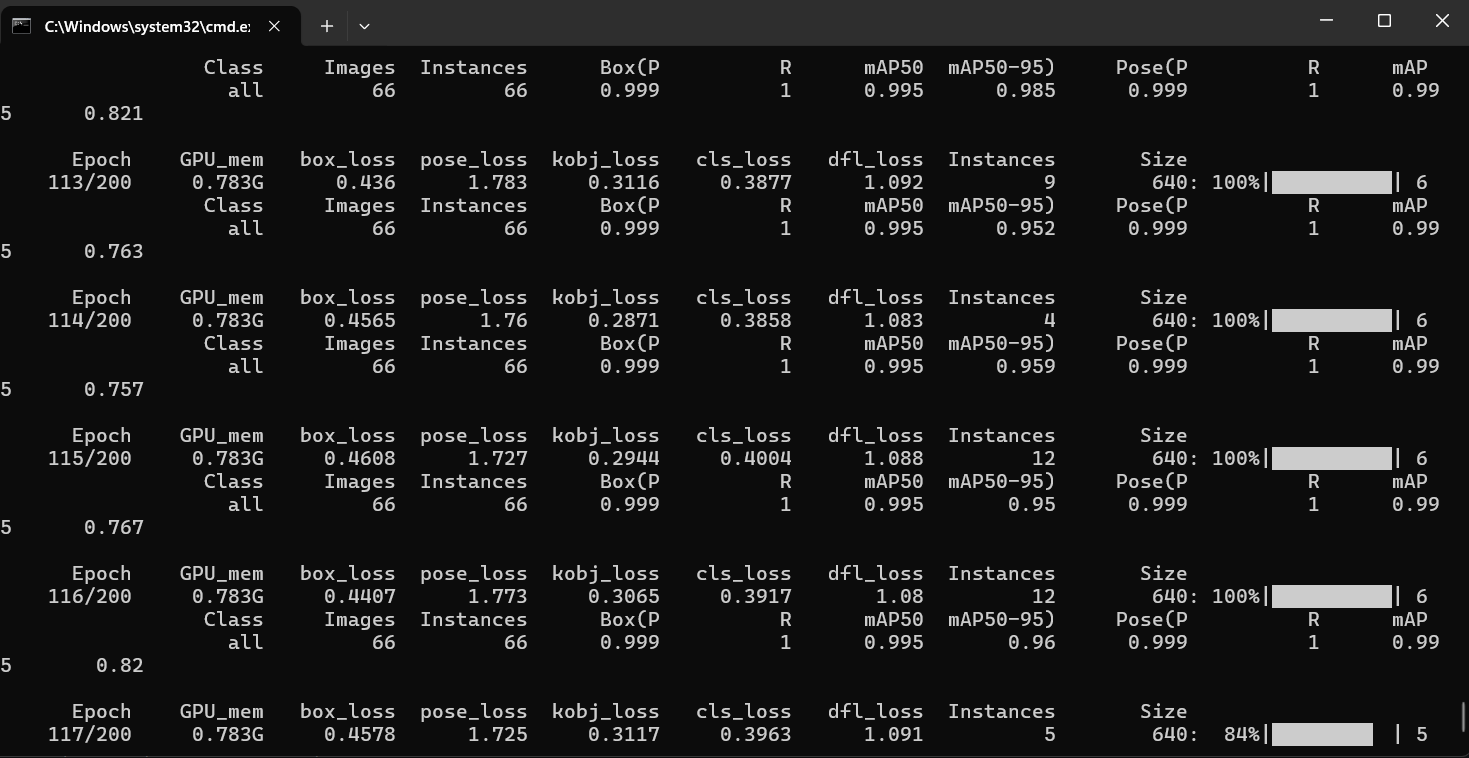


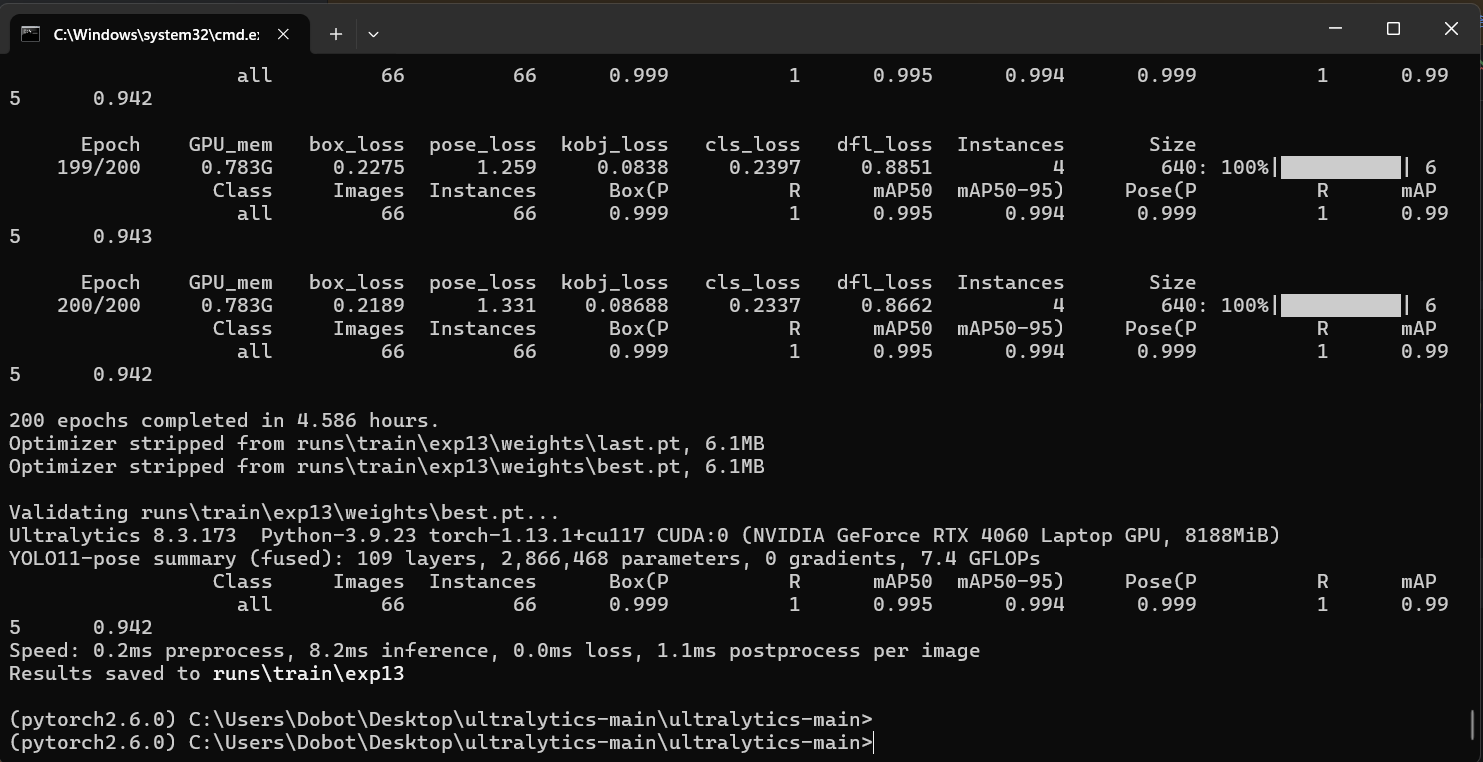
在kaggle平台的front\_squat\_data29数据集上进行训练：

自制用于训练的yaml文件：

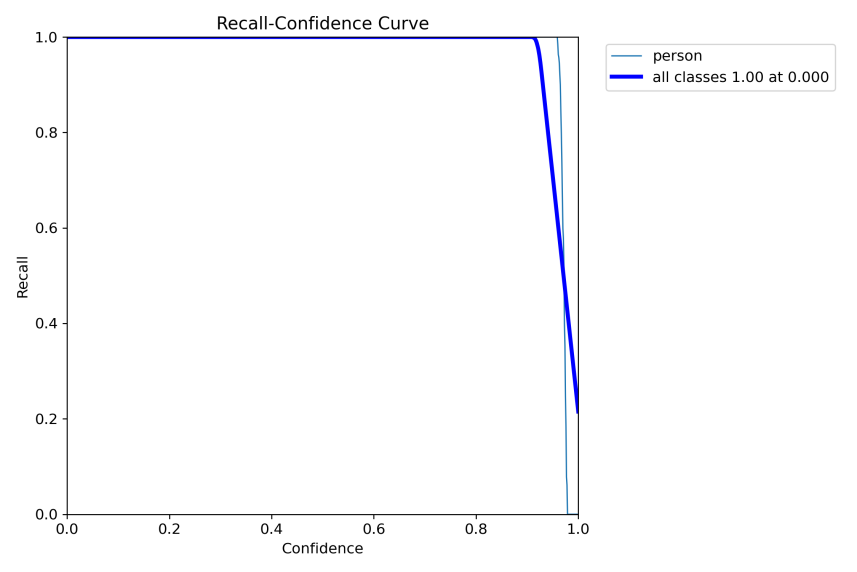


训练中：

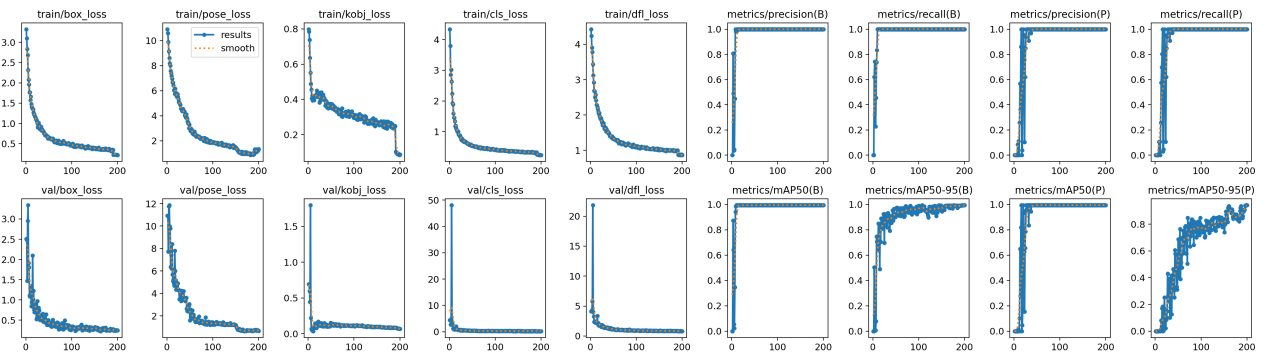




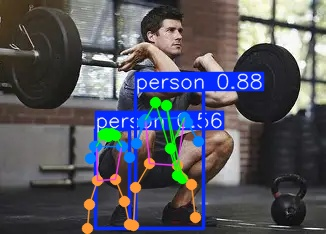
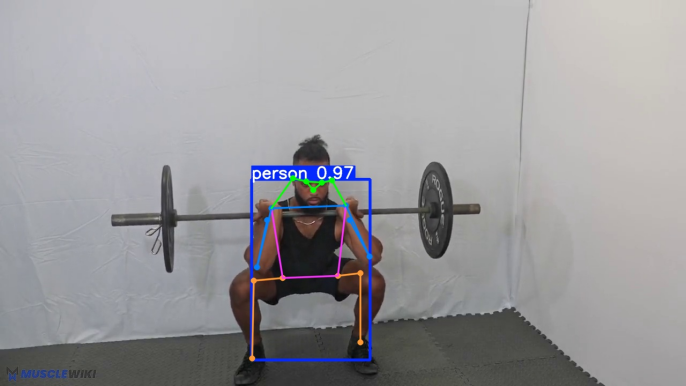
训练、验证结果：





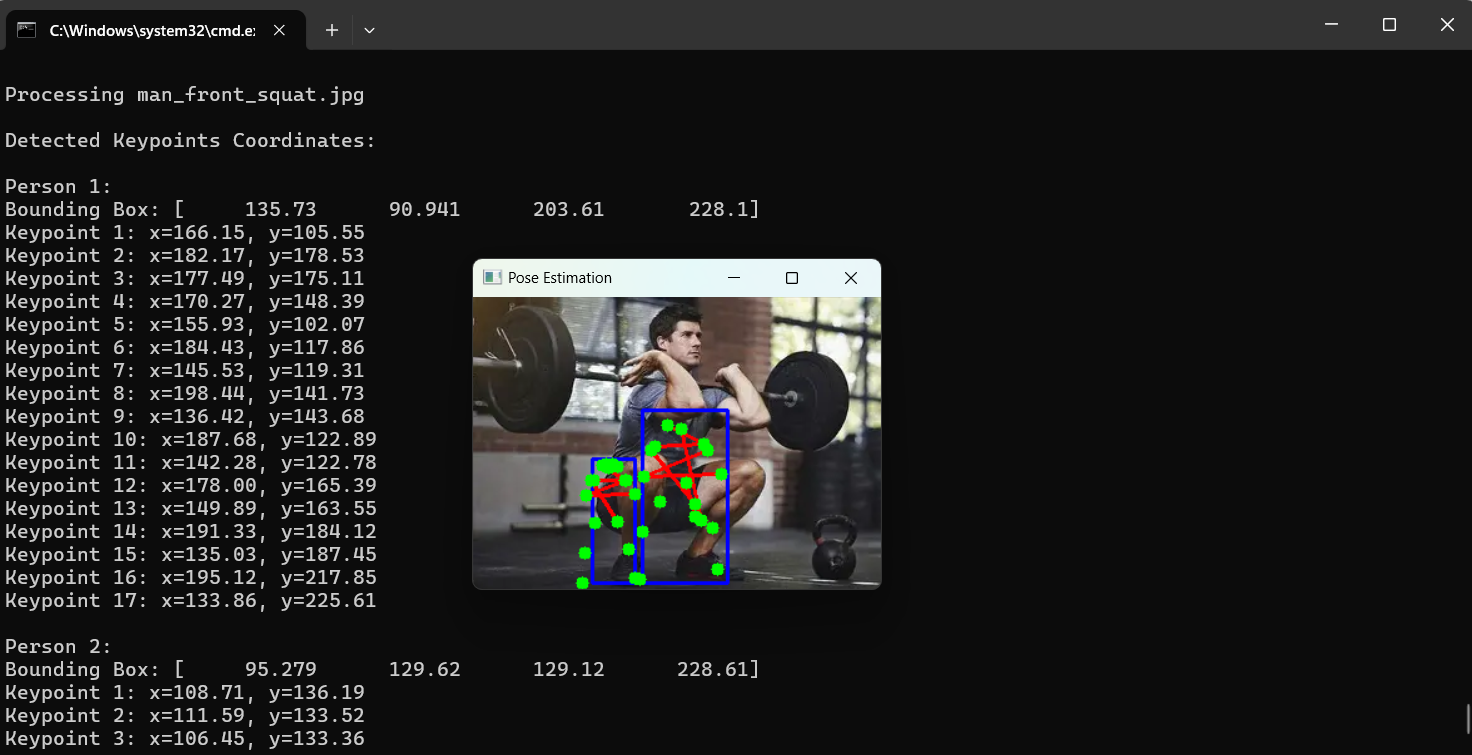


测试：



17个关键点，3维坐标，1个类别

打印测试图片中关键点的坐标：



Ultralytics YOLO Transformer：

借鉴了Vision Transformer（ViT）、Deformable DETR、PaddleDetection

Ultralytics YOLO Transformer定义了以下模块：

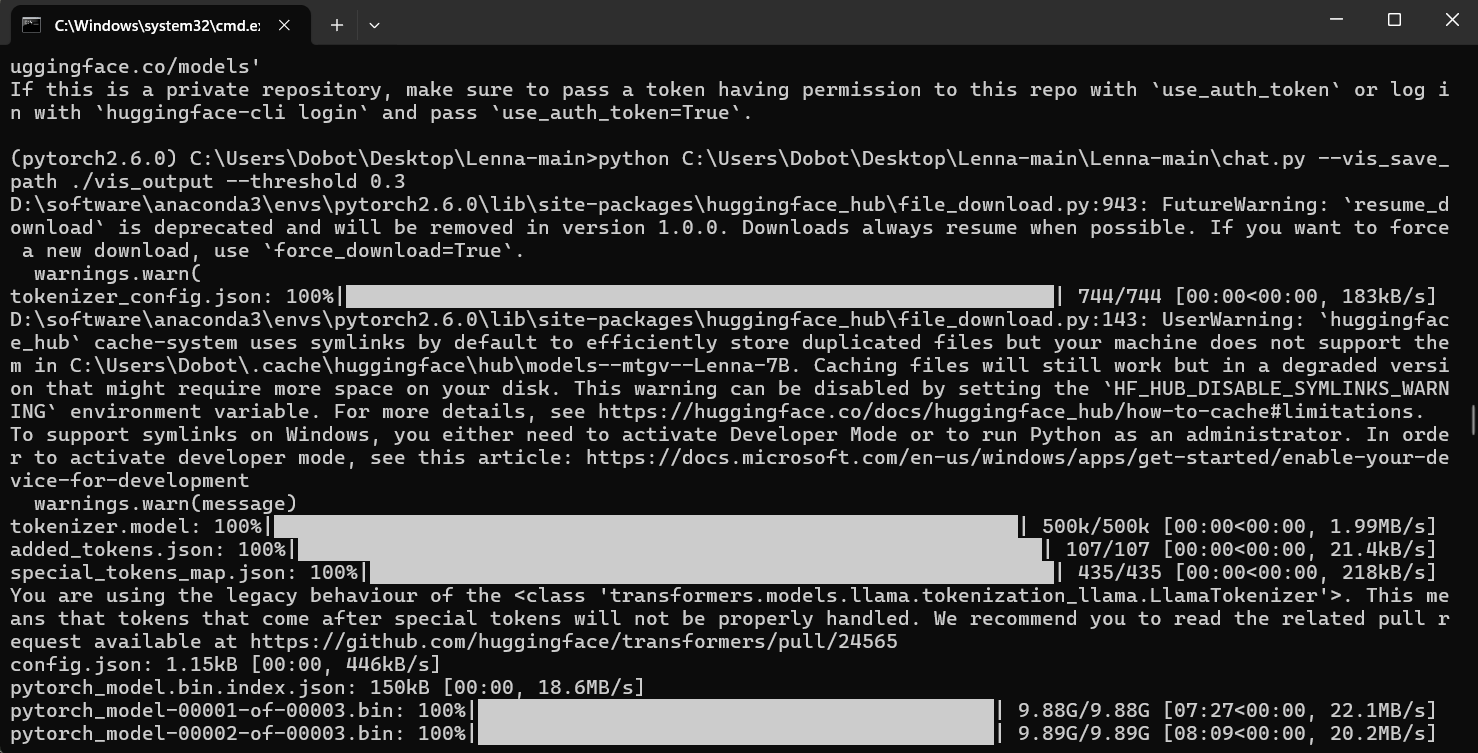
| **类名** | **功能** |
| --- | --- |
| TransformerEncoderLayer | 标准 Transformer Encoder 层 |
| AIFI | 2D 图像专用的 Transformer Encoder 层，带 2D 正弦位置编码 |
| TransformerLayer | 简化版 Transformer 层（去掉了 LayerNorm） |
| TransformerBlock | Vision Transformer Block，支持 2D 输入 |
| MLPBlock | 多层感知机模块 |
| MLP | 通用 MLP 网络 |
| LayerNorm2d | 2D 版本的 LayerNorm |
| MSDeformAttn | 多尺度可变形注意力模块（来自 Deformable DETR） |
| DeformableTransformerDecoderLayer | 可变形 Transformer 解码器层 |
| DeformableTransformerDecoder | 完整的可变形 Transformer 解码器 |

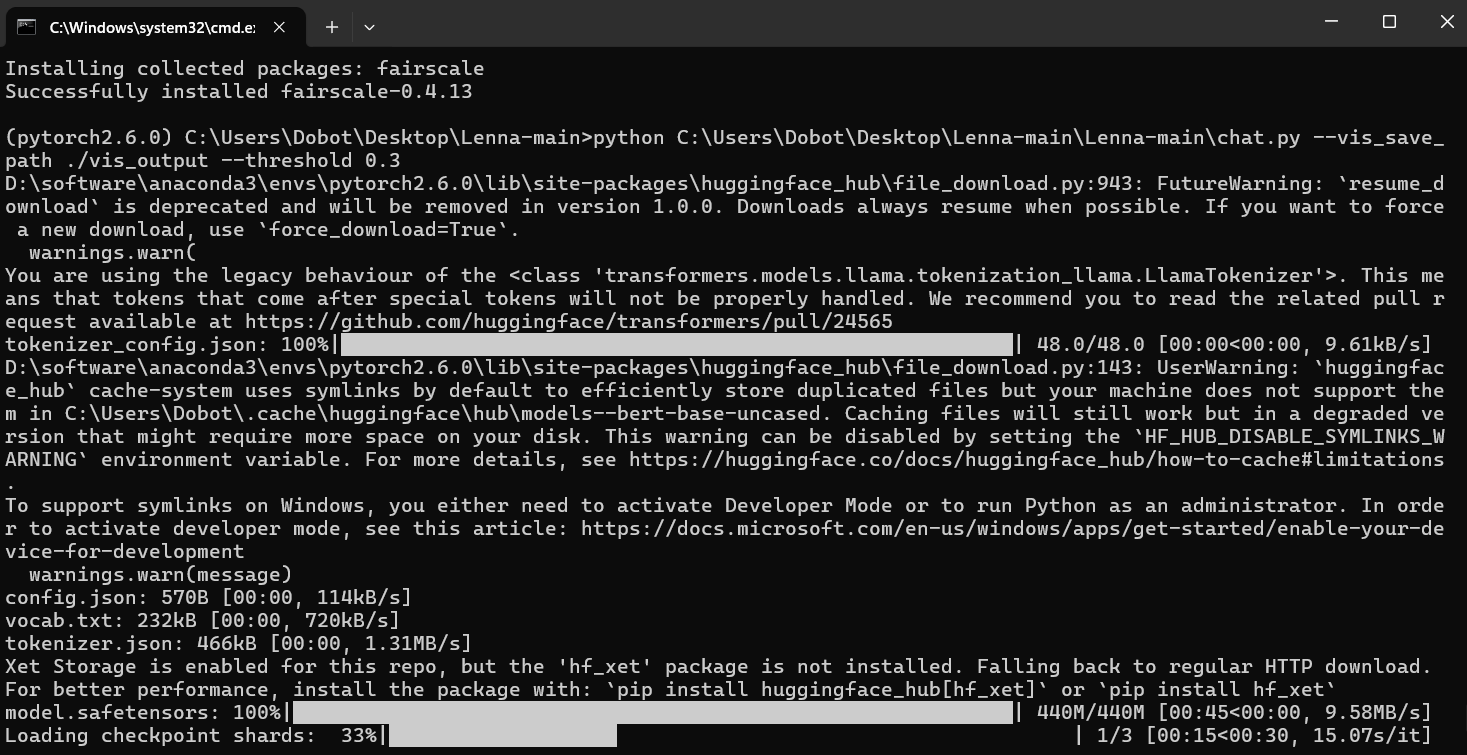
参考教程：<https://developer.d-robotics.cc/rdk_doc/Quick_start/>

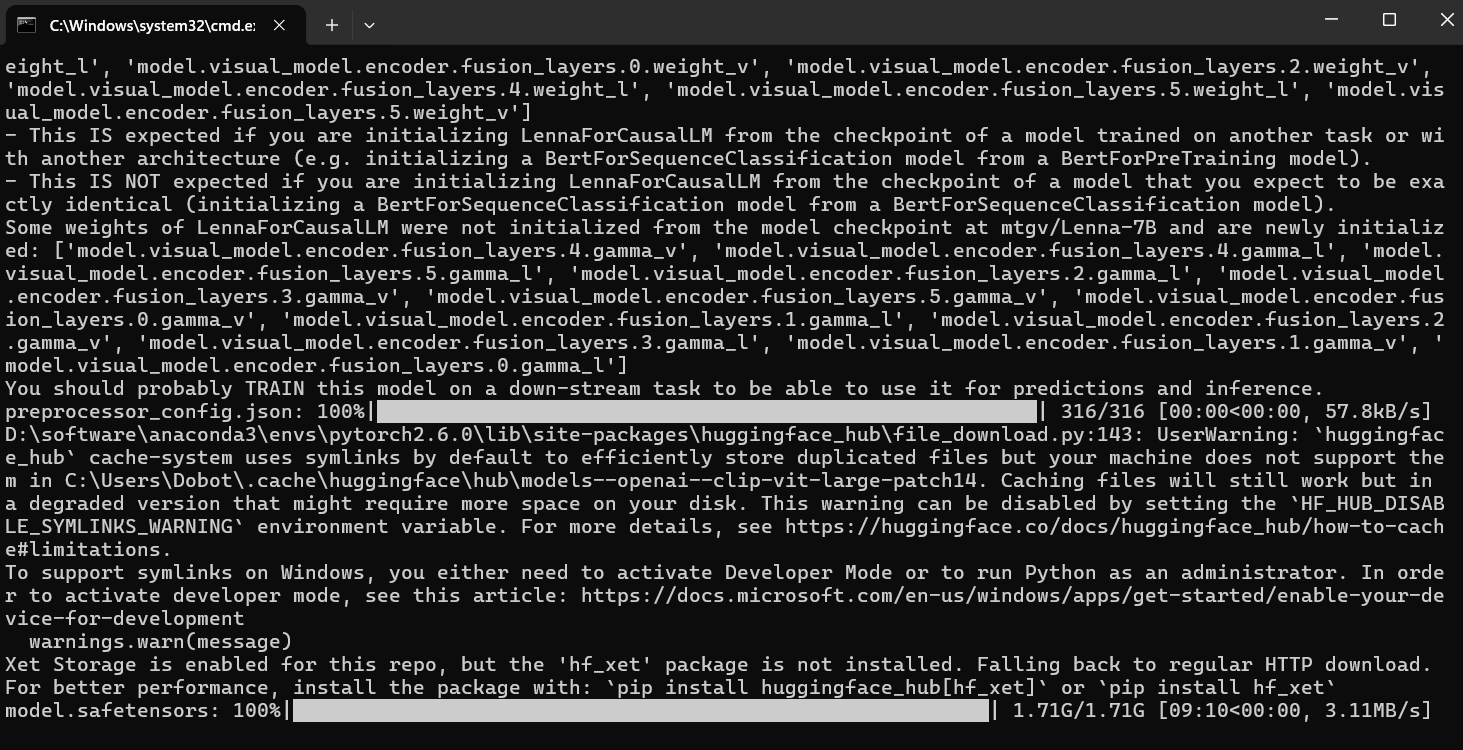
关键点：将pth模型转化为边缘芯片兼容的格式

1. 部署视觉语言模型：Lenna-7B

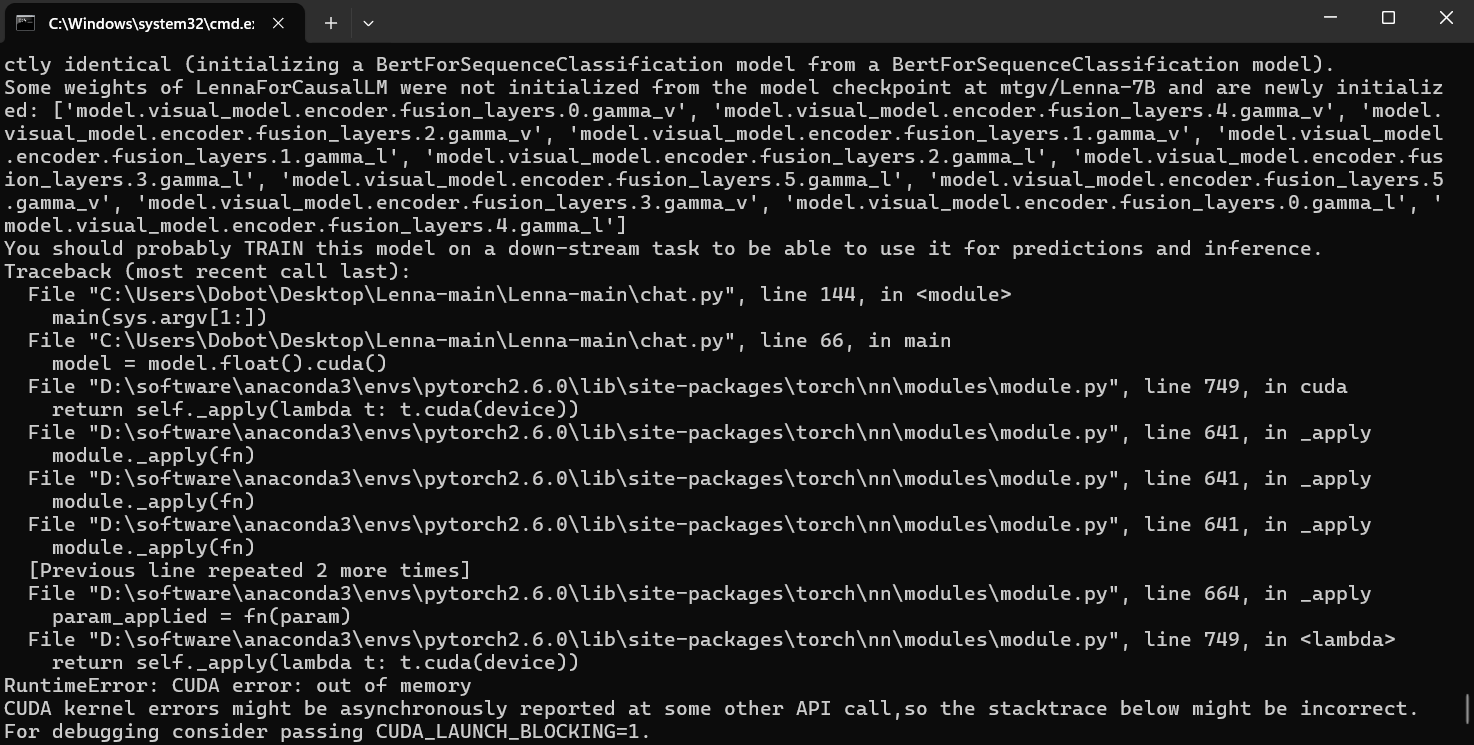
参考资料：<https://github.com/Meituan-AutoML/Lenna>

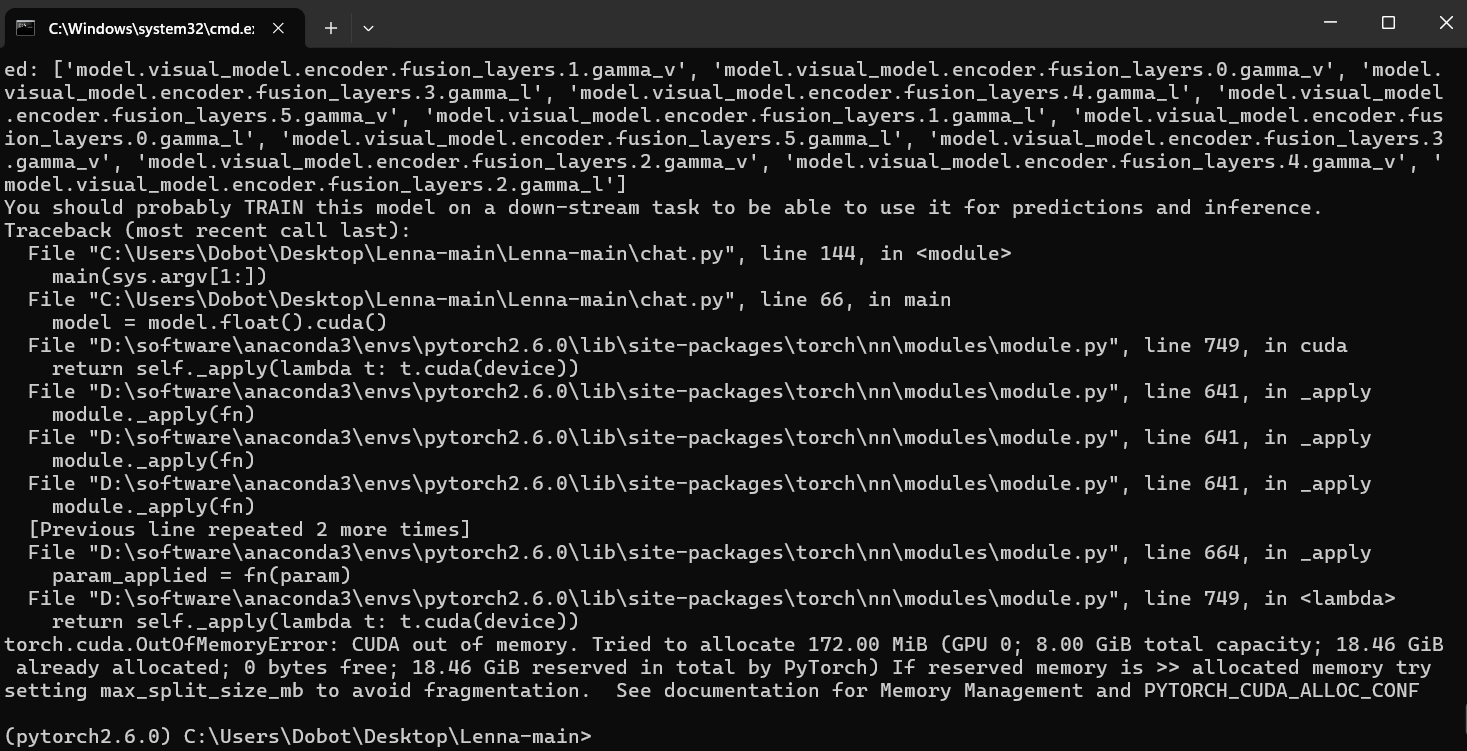


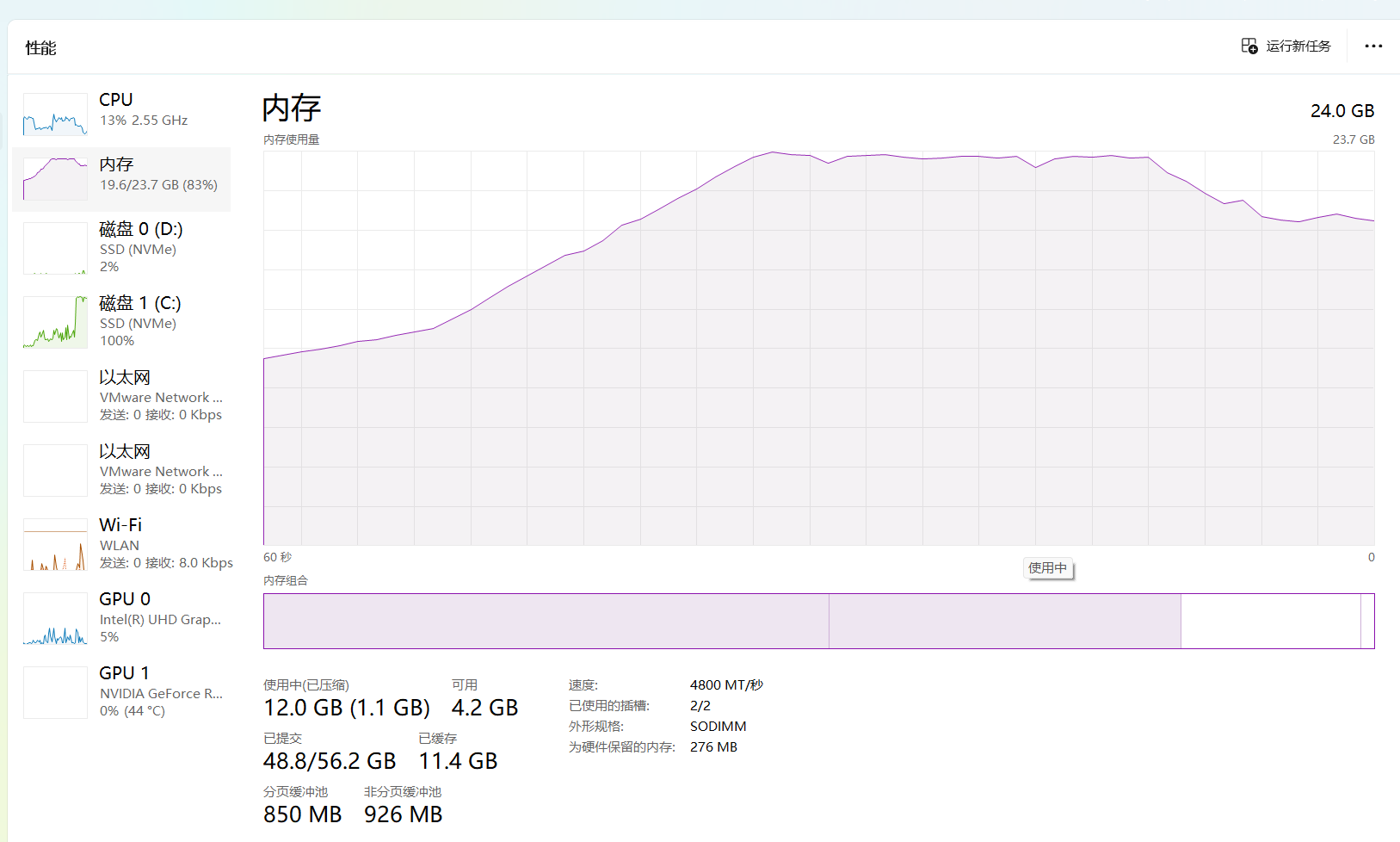


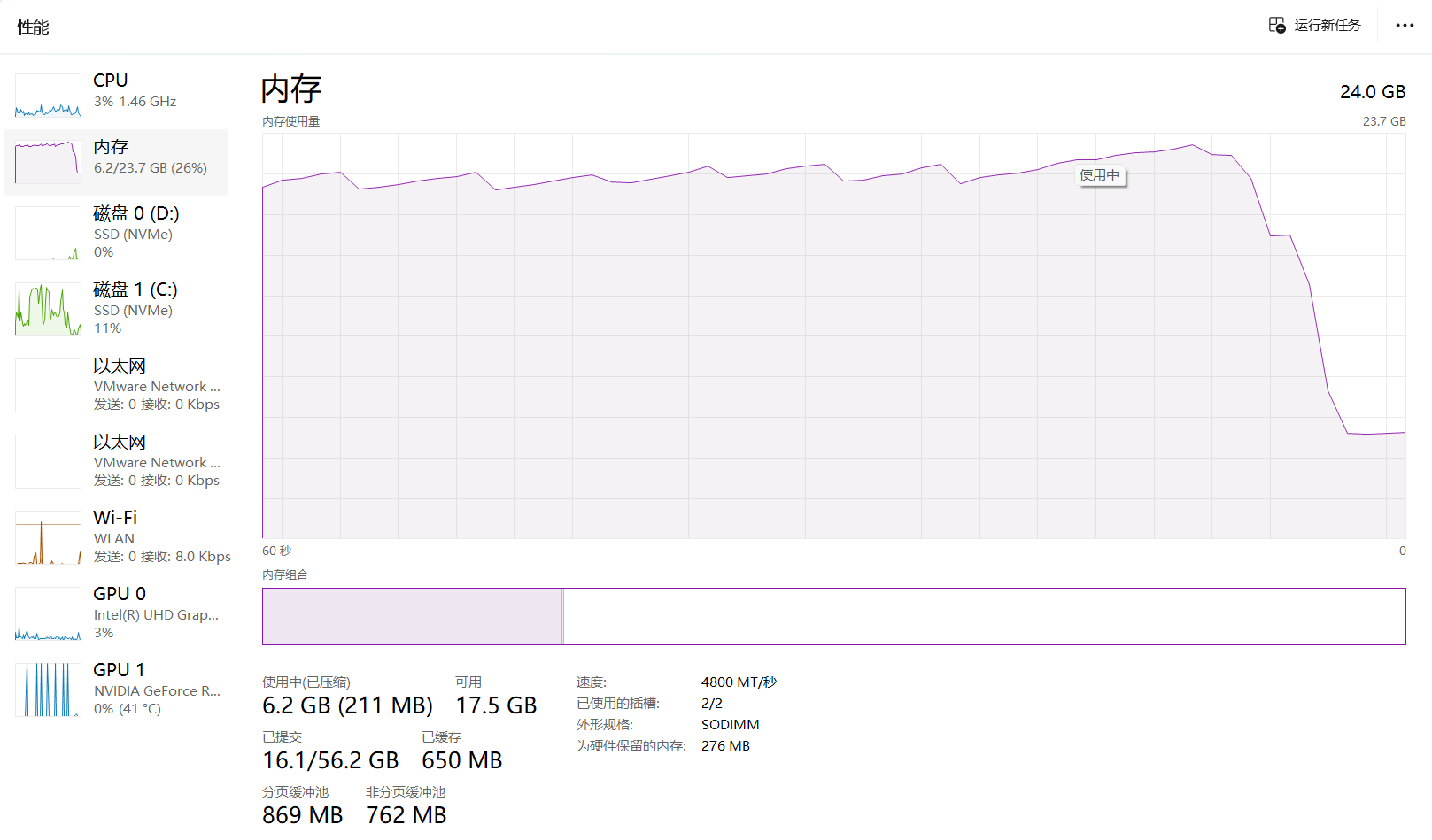


遇到问题：显存溢出









解决方案：

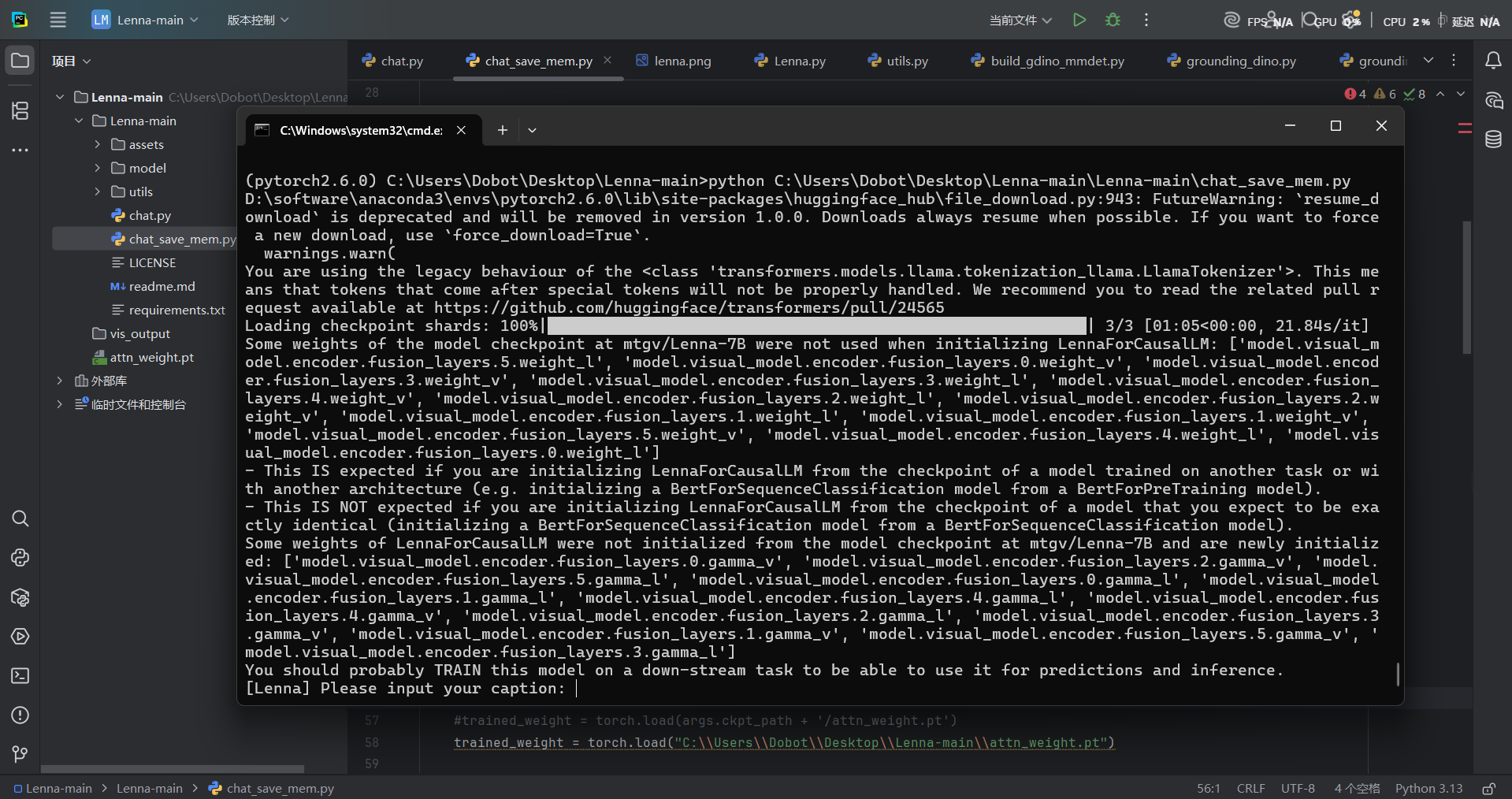


训练前释放显存



“torch\_dtype”:torch.float16:使用半精度浮点数加载模型参数；

“low\_cpu\_mem\_usage”:True:启用HuggingFace的低CPU内存加载模式；



加载权重成功，成功启动模型

### 下周计划：按照教程烧录SDK ×5系统，将YOLO11-pose算法经过量化后部署到SDK ×5上

### 其他操作：

### 1. ****模型结构优化与改进****

**轻量化设计**：尝试剪枝、量化或知识蒸馏，减小模型体积，提升推理速度，方便部署到边缘设备或移动端。

**融合新技术**：结合Transformer、注意力机制等最新架构，提升关键点检测精度。

**多任务学习**：融合人体姿态估计和行为识别，或者增加姿态跟踪功能，实现更丰富的应用。

### 2. ****数据增强与扩充****

**自动数据增强策略**：用AutoAugment、RandAugment等自动策略提升训练数据多样性。

**合成数据**：结合模拟数据或GAN生成数据，扩充样本，提升模型鲁棒性。

**多模态数据融合**：结合视频、IMU传感器数据等，提高姿态估计准确性。

### 3. ****算法优化与加速****

**混合精度训练**（FP16）：减少显存占用，加快训练速度。

**自适应推理**：根据场景复杂度动态调整推理精度和速度。

**分布式训练**：利用多GPU、多节点，提升训练效率，解决大规模数据训练瓶颈。

### 4. ****应用创新****

**实时动作识别**：基于姿态估计做实时动作分析，拓展应用场景。

**异常行为检测**：结合姿态估计，实现异常行为预警，比如跌倒检测。

**跨领域迁移**：将模型应用到体育分析、康复医疗、动画制作等领域，结合实际需求进行定制。

### 5. ****系统集成与优化****

**端到端自动化训练和部署流水线**：包括数据准备、训练、验证、打包部署，实现持续集成（CI/CD）。

**模型可解释性分析**：用可视化工具解释模型决策，提升系统可信度。

**容错机制设计**：应对姿态检测失败或遮挡，设计多模型融合或后处理方案。

### 6. ****研究新挑战点****

**弱监督或无监督学习**：减少对大量标注数据的依赖。

**多人体复杂场景的鲁棒性提升**：处理遮挡、交叉、多视角融合。

**3D姿态估计扩展**：从2D关键点提升到3D人体姿态估计，提高应用层次。