

U-Net 学习报告

作者：你的姓名

2025 年 5 月 14 日

1. 研究背景与模型简介

U-Net 是一种专为生物医学图像分割设计的卷积神经网络架构。自 2015 年提出以来，因其优异的定位能力和端到端的训练方式，被广泛应用于组织、器官、细胞等医学图像的精确分割任务。其核心特征是“编码器-解码器”结构，并通过跳跃连接融合多尺度特征信息，有效提升了分割精度。

2. 理论学习过程中的关键问题与理解

2.1. 模型结构认知难点

初学时对“对称结构”、“跳跃连接”、“上采样与下采样”的含义及其作用理解不清。通过阅读原始论文 *U-Net: Convolutional Networks for Biomedical Image Segmentation*，配合 PyTorch 代码结构图，深入理解编码器提取语义信息，解码器恢复空间结构，跳跃连接用于细节保留。

2.2. 损失函数选择

对 Binary Cross Entropy、Dice Loss、IoU 等评价指标之间的关系不明确。通过查阅相关文献，确认在二分类掩码任务中使用 Dice Loss 更适合，IoU 可用于评价。

2.3. 正则化结构理解

如 Batch Normalization 与 Dropout 的作用在小样本下是否有疑惑。阅读相关论文与实测发现，小数据集下 Dropout 可省略，BN 的作用有限。

3. 代码实现与调试过程中的问题总结

3.1. W&B 网络配置失败

训练中 WandB 报错 `ProxyError`，无法同步日志。采用 `wandb.init(mode="offline")` 方式成功解决，待网络恢复后可同步。

3.2. 内存占用过高

训练中经常出现 CPU 内存达到 99%，程序中断。通过：

- 增加 Windows 虚拟内存至 19GB；
- 减小 `num_workers`；
- 降低图像尺寸；

成功缓解内存瓶颈。

3.3. 尺寸不一致导致拼接失败

模型拼接时出现 `torch.cat` 报错，因上采样后的特征图尺寸与跳跃连接不一致。通过在 `Up.forward()` 中添加中心裁剪逻辑修复：

```
x2 = x2[:, :, diffY//2 : -(diffY - diffY//2), diffX//2 : -(diffX - diffX//2)]
```

3.4. 损失函数维度不匹配

模型输出通道为 2，但目标 `mask` 为 1，导致 BCE 报错。修复方法是将输出通道改为 1，并确保使用 `sigmoid` 激活后再传入 BCE loss。

3.5. GPU 未启用

虽然硬件支持 GPU，但运行时报错 `Torch not compiled with CUDA enabled`。排查发现安装的是 CPU 版 PyTorch，通过以下命令更换：

```
pip install torch torchvision torchaudio --index-url https://download.pytorch.org/whl/cu121
```

4. 学习收获与后续计划

本次 U-Net 的理论学习与工程实现过程帮助我深入理解了语义分割模型的结构原理、调参方法、训练监控与资源管理。掌握了：

- 编码-解码结构及跳跃连接的实际意义；
- 多种损失函数的选择与应用；
- GPU 配置与 W&B 日志工具的完整使用流程；
- 常见训练报错的排查与修复流程。

后续学习方向：

- 探索 Attention U-Net、R2U-Net 等增强结构；
- 学习多类语义分割任务（如器官级别标注）；

- 尝试集成 SAM 或 Segment Anything 作为后处理增强；
- 构建完整的医学图像分割部署系统。

本报告总结了本人阶段性学习成果，并为后续研究与工程开发打下基础。