### **报错核心原因**

* ****模块缺失问题****：代码在执行过程中，从 main.py 文件的第 27 行开始，尝试从 engine.py 文件（位于 /root/NeuTRENO\_NeurIPS2024/DeiT/ 路径下）导入 train\_one\_epoch 和 evaluate 函数，而在 engine.py 文件的第 9 行又尝试从 fvcore.nn 导入 FlopCountAnalysis ，但系统提示找不到 fvcore 这个模块，导致整个程序无法正常运行下去。每次启动的多个进程（不同 local\_rank 的进程）都会因为这个导入问题而报错退出，比如 local\_rank 为 0、1、2、3 的进程（对应的 pid 分别为 21383、21385、21386、21387 等不同尝试中的对应进程）都以 exitcode: 1 的状态失败。

### **相关衍生报错及情况说明**

* ****分布式训练相关报错****：由于最初的模块导入失败，进而引发了一系列与 torch.distributed.elastic 相关的报错，这是因为分布式训练框架检测到了子进程运行失败，不断地尝试重启工作进程组（Worker group）来重新运行任务，最多会尝试 3 次（从相关的报错信息中可以看到如 1/3 attempts left 等表示剩余尝试次数的提示），但每次因为根本的模块缺失问题依旧存在，所以一直重复报错失败。例如有诸如设置回复文件（reply file）相关的信息记录，以及不同阶段的分布式训练状态（如 Rendezvous'ing worker group 表示协调工作进程组等操作）的日志输出，都是围绕着整体运行失败而产生的连锁反应。

### **解决建议**

1. ****安装**fvcore**模块****：使用合适的 Python 包管理器来安装 fvcore 模块，如果你使用 pip ，可以在命令行中（确保你的环境激活正确，比如使用了 conda 等环境管理工具的话，要激活对应的包含此项目代码运行环境的 conda 环境）输入以下命令安装：

收起

bash

pip install fvcore

如果使用 conda ，可以尝试：

收起

bash

conda install -c conda-forge fvcore

1. ****检查环境一致性****：确认你安装 fvcore 的 Python 环境就是运行代码（main.py 等相关代码）的 Python 环境，有时候可能存在多个 Python 环境，容易导致安装的模块在实际运行代码的环境中不可用的情况。
2. ****代码路径和依赖检查****：再次检查 engine.py 文件中导入 fvcore.nn 的语句是否正确书写，确保没有拼写错误等语法层面的问题，同时确认整个项目代码所依赖的其他模块和库是否都正确安装和配置，避免存在其他潜在的依赖冲突或者缺失情况，因为有可能存在因为某个依赖缺失导致连锁反应，看起来只是 fvcore 导入失败，但实际可能还有相关联的其他问题。

解决了 fvcore 模块缺失这个核心问题后，再重新运行代码，理论上应该可以避免当前出现的这些反复报错情况，使程序能够正常进入后续的训练等流程（前提是其他部分代码逻辑和配置也都是正确的）。











