作业 3：AlphaZero

清华大学软件学院
软件工程探索与实践（人工智能板块），2023 年春季学期

1 介绍

本次作业需要提交说明文档（PDF 形式）和 Python 的源代码。注意事项如下：

- 本次作业满分为 100 分，并有一道附加题，若得分超过 100 分，则按照 100 分截断。
- 作业按点给分，因此请在说明文档中按点回答，方便助教批改。
- 请不要使用他人的作业，也不要向他人公开自己的作业，否则将受到严厉处罚，作业分数扣至-100（即倒扣本次作业的全部分值）。
- 统一文件的命名：{学号}_{姓名}_hw3.zip

2 网络与训练例程

本次作业是期末大作业的训练模块。在本题中，你需要在给出的代码框架与你之前两次作业实现的模块基础上实现 9×9 棋盘上针对围棋的 AlphaZero 算法的训练与评测。本题需要完成以下内容，提交代码和实验报告，代码见 `./code`。

1. 补全 GoNNet 类，设计并实现围棋的特征提取网络，你可以参考代码演示中的网络结构，但需要做出自己的改动。
2. 补全 GoNNetWrapper 类，实现损失函数的计算、反向传播与更新，损失函数的形式请参考 AlphaZero 原论文公式 1 的描述[1]。
3. 适当修改 MCTS 类，实现 PUCT，并使之可以支持神经网络对于叶子节点价值的评估，请参考 AlphaZero 原论文补充材料的对于 MCTS 的描述[2]。
4. 代码框架在生成自我对弈对局时使用了 AlphaGo Zero 的策略。补全 Trainer 类，根据 AlphaGo Zero 的自我对弈策略更新最优模型与 baseline 模型，AlphaZero 原论文中第二页提到了其与 AlphaGo Zero 在自我对弈策略上的区别。
5. **（附加题）**对比两种自我对弈策略在训练效果与训练稳定性上的区别。（5pt）
6. 训练模型并撰写实验报告，在报告中画出你最终实现的网络架构图（可以参考 ResNet 原论

[1] `https://www.science.org/doi/10.1126/science.aar6404`

[2] `https://www.science.org/doi/suppl/10.1126/science.aar6404/suppl_file/aar6404-silver-sm.pdf`

文[3]中 Figure 3 的表现形式)，汇报先手/后手对 RandomPlayer 的胜率随训练时间的变化，模型更新频率，损失函数曲线。你需要在最终提交的作业中附上你的模型，模型最长的训练时间不超过 3 天。

提示:

1. 训练入口位于 main.py，训练例程位于 train_alphazero.py，动手之前，请仔细阅读代码中的注释，确保你已了解问题定义和代码框架。本次作业代码工作量不大（< 100 行），重点在于理解 AlphaZero 这一深度学习系统的框架。
2. 本次作业中大部分的分值取决于你的实现正确性，少部分的分值取决于你与同学对弈的结果，如果你的训练结果可以做到对 RandomPlayer 有 100% 的胜率，就可以拿到大部分的分数。
3. 在使用 GPU 训练之前，你应当首先完成各个模块独立的测试，然后在 CPU 上完成整个训练流程的调试与测试，在 GPU 上调试的难度会很高。提前测试可以防止代码错误积重难返。
4. 请确保 MCTS 类的对外接口与助教给出的代码框架一致，如果因为接口原因无法与同学对弈，则会失去对弈部分的全部分数。
5. 如果需要计算资源支持，可以使用 Anylearn 机器学习管理系统[4]，exp_anylearn.py 中已经实现了简单的快速启动实例，你可以参考用户文档[5]，SDK 文档[6]，CLI 教程[7]修改训练启动代码。

[3] https://arxiv.org/pdf/1512.03385.pdf

[4] https://anylearn.nelbds.cn

[5] https://anylearn.feishu.cn/file/C8tabHqoeoNtNwxAS9vctWPAnXd

[6] https://thulab.github.io/anylearn/

[7] https://thulab.github.io/anylearn/cli/tutorial.html