开悟比赛-XJTU 太初队技术整理与分享

王宇航 西安交通大学人工智能学院

戴洋 西安交通大学人工智能学院

王思哲 西安交通大学人工智能学院

石昊忱 西安交通大学人工智能学院

寇谦 西安交通大学人工智能学院

指导老师:兰旭光

一、 简介

在 2022 年 9 月-2023 年 4 月举办的腾讯第三届开悟 MOBA 多智能体强化学习大赛中，我们队伍(XJTU 太初)在初赛中有幸获得了第 3 名的成绩。这得益于实验室老师与腾讯官方的大力支持， 也得益于我们队伍本身的技术探索与积累。本文首先会简单叙述比赛的基本情况，随后从各关键模块出发，简要介绍本队伍在开悟比赛中的探索历程与心得体会。

二、 参赛概况

本次比赛参赛的五名队员来自兰旭光教授的研究生团队，对强化学习在游戏 AI 领域的应用兴趣浓厚。兰教授的团队在上届比赛取得了第三名，这届比赛更换了新一届的队员，希望能保持上届比赛的发挥水平。在时间投入方面，队员们在科研学习的同时参与比赛，在不同比赛阶段用时不同，在开发阶段集中开发，在训练阶段只留小部分时间调参。在参赛过程中，每名队员在代码开发，算法、特征、奖励设计等方面均有所贡献，由于队员们大多第一次参赛，所以在初赛过程中以熟悉开悟环境和架构为主，对强化学习方法的探索较少。

三、网络设计

本次比赛的神经网络参考论文[1]中的网络设置，由 CNN 和 MLP 处理不同特征，送入 LSTM，后接 value/policy head。我们的 Value head 网络参考论文[2]中的 multi-head value 设计，并对 soldier、organ 等 unit-based 特征做 attention+maxpool 处理，对 LSTM 做 layernorm 处理，将部分时序无关的特征 bypass 到 LSTM 网络之后，使用 NoisyNetwork[3]替代 MLP 做 policy head 以辅助探索。我们曾尝试用 GRU 代替 LSTM，发现效果并没有太大差异，所以并未采用。

四、奖励体系

我们在本次比赛中并未对奖励函数做修改，在训练中直接应用 baseline 默认设置。

五、特征与规则

我们在本次比赛并未对原始特征进行改动，并未基于规则实现前后处理等操作。

六、 强化学习算法

我们使用[1]中的 dual-clip ppo 算法，用 multi task 的方式同时训练所有环境 settings。我们也尝试了为每个英雄单独训练一个模型，再通过模型蒸馏整合到一个模型的方案，但在实现后我们发现这个方案对于算力的要求较大，增大了调参试错成本，所以最后并未采用这个方案。

七、 系统工程架构

我们对模型大小和 batchsize 大小进行了调整以更合理地使用开悟平台的算力配置。对 actor 代码做了更改以实现在训练和评价时从指定的对手模型池中采样，这样能够让我们使用一个更好的模型引导其他模型初期的训练，也能让我们在训练时从监控面板中得到更有用的信息。我们在消融实验中发现使一定比例的更好的模型作为对手采样能够加速模型的训练过程，这或许可以作为训练后期回退调参时加速训练的办法，但实际上在后期训练时时间较紧，我们没有太多的回退调参的机会，因此未采取这项功能进行训练。

八、 模型迭代过程

在模型迭代中，我们逐渐降低学习率和 entropy loss 系数。并尝试对 ppo clip ratio 和 gamma 进行调整，但并未从中发现明显规律，由于算力紧缺，我们在模型迭代的途中对参数进行动态调整，并未做消融实验，所以对其具体影响本身不可知。

九、 训练效果分析

我们在模型蒸馏方案、网络模型实现、训练和评估模型池三个部分投入了主要精力开发。最后模型蒸馏方案被舍弃，训练和评估模型池在训练调参过程中对我们有所帮助，而基于网络模型的单模型多阵容 multi-task 训练方案被我们采用。在模型迭代训练前期，我们耗费了一些算力对我们的代码实现作了测试，同时对一些网络设计的方案作了消融实验。在确认最终方案后，我们开始从头进行模型的迭代训练，途中会因为调参不当造成模型过早收敛或者训练效果不佳，则会回退再开始训练。从调参角度，学习率和 entropy loss 系数的降低对模型水平的提升是明显的，而 gamma 和 ppo clip rate 对模型训练的影响也很明显，但是其影响并非单调，在不同训练阶段的效果也不同，所以很难衡量其效果。最终我们得到的模型水平不算很高，会犯一些从人类角度较为明显的决策错误。

十、 总结与展望

在初赛阶段，由于队员们花费较多时间熟悉开悟平台和开发流程，所以对算法方面开发实验较少，最后靠一些网络改进和调参取得了一定效果。囿于初赛的 tensorflow1 代码框架，对于需要对计算图结构、网络参数或者梯度进行灵活操作的新兴算法的开发效率较低。在复赛支持 torch 开发之后，我们会尝试类似 self tuning actor critic 的在 IMPALA 架构上有一定效果的算法尝试增加训练效率，同时对模型为何收敛在一个较低的水平做探究，或许这是因为网络 capacity 耗尽，或许是在自对弈情况下策略提升路径过于单一，最终收敛在了局部最优上，也有可能其他原因。

参考文献

[1] Ye, D., Liu, Z., Sun, M., Shi, B., Zhao, P., Wu, H., ... & Huang, L. (2020, April). Mastering complex control in moba games with deep reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 34, No. 04, pp. 6672-6679).

[2] Ye, D., Chen, G., Zhang, W., Chen, S., Yuan, B., Liu, B., ... & Liu, W. (2020). Towards playing full moba games with deep reinforcement learning. *Advances in Neural Information Processing Systems*, *33*, 621-632.

[3] Fortunato, M., Azar, M. G., Piot, B., Menick, J., Osband, I., Graves, A., ... & Legg, S. (2017). Noisy networks for exploration. *arXiv preprint arXiv:1706.10295*.