开悟比赛-SIAT 嘎嘎上分队技术整理与分享

李璇 中科院深圳先进技术研究院数字所

马珂 中科院深圳先进技术研究院数字所

吴都 中科院深圳先进技术研究院数字所

程鹏杭 中科院深圳先进技术研究院数字所

指导老师:孟金涛

一、 简介

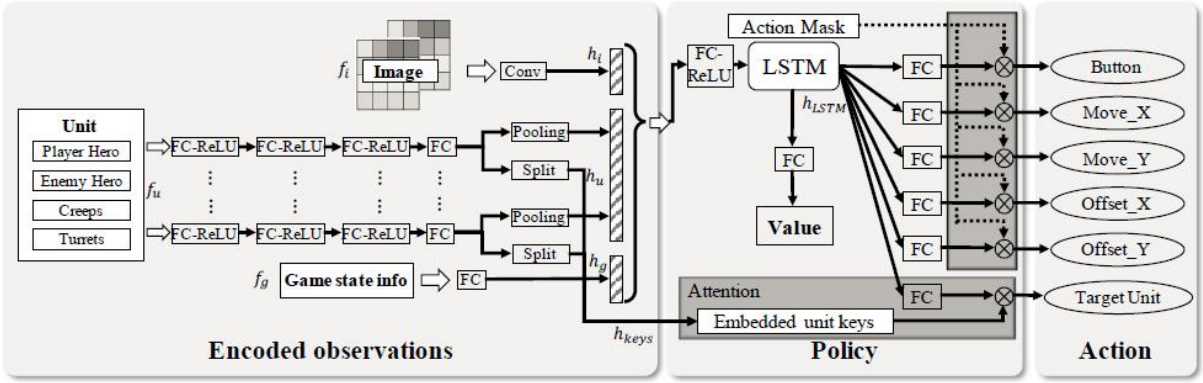
在 2022 年 9 月-2023 年 4 月举办的腾讯第三届开悟 MOBA 多智能体强化学习大赛中，我们队伍(SIAT 嘎嘎上分队)在初赛中有幸获得了第 4 名的成绩。这得益于实验室老师与腾讯官方的大力支持， 也得益于我们队伍本身的技术探索与积累。本文首先会简单叙述比赛的基本情况，随后从各关键模块出发，简要介绍本队伍在开悟比赛中的探索历程与心得体会。

二、 参赛概况

我们队伍共有 4 名成员，均为中国科学院深圳先进技术研究院数字所高性能中心的硕士生，队伍中有热爱王者荣耀的同学，也有做强化学习方向的同学，出于对游戏 AI 的好奇和对提升自身实践能力的渴望，在中心孟老师的组织下，我们来自同一个中心四个课题组的四位同学组成了这支队伍。初赛阶段，每周花费 1-2 天时间，每两周一次小组会交流想法和比赛进展。

三、网络设计

网络结构我们沿用了官方提供的默认网络模型，结构大致如下图：



图一 网络结构

与图中并不完全一样，或许是出于资源的考虑，这里并没有用到图像特征f_i。

该网络结构 work 的一些创新点在于：首先，在该网络中设计了目标注意力机制，以帮助 agent

在 MOBA 战斗中选择目标。其次，英雄利用 LSTM 学习技能组合，这对于立即造成致命伤害至关重要。第三，进行控制依赖关系的解耦以形成多标签近端策略优化（PPO）目标函数。第四，开发了一种基于游戏知识的修剪方法，称为“动作屏蔽”(action mask)，以指导强化学习过程中的探索[1]。

四、奖励体系

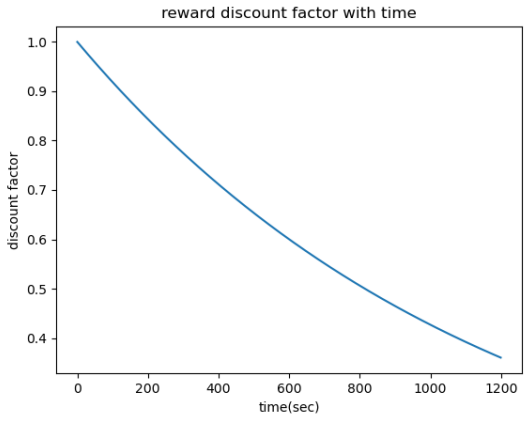
观察录像，主观判断模型的不足之处，提高训练前中后期不同阶段对应项的系数。具体地，以表一中默认系数为基础，先调高经济、经验、血量等稠密项系数，让 agent 能学会快速发育，期望 agent 能够学会抢二级，但可能训练时间过短，没能达到这个预期。之后，再将稠密奖励项系数调低，将击杀和死亡的系数调高，让 agent 能够在发育的基础上提高 KDA，多杀人。

表一 奖励函数各项权重

Reward	Weight	Type	Description
hp_point	2.0	dense	the rate of health point of hero
tower_hp_point	10.0	dense	the rate of health point of tower
money (gold)	0.006	dense	the total gold gained
ep_rate	0.75	dense	the rate of mana point
death	-1.0	sparse	being killed
kill	-0.6	sparse	killing an enemy hero
exp	0.006	dense	the experience gained

根据游戏经验，前期发育比较重要，且因为英雄能力变化，为获取同等 reward，前期所需时间更多也更难，后期 reward 会膨胀的太多，所以给 reward 加上一个时间的折扣，越往后折扣越大， reward 随时间衰减，以提高前期 reward 的权重，平衡游戏的前中后期 reward，衰减周期为 10 分钟，具体计算公式如下：（这部分的折扣与强化学习算法中的折扣因子的作用和意义不同）

$$reward_i = reward_i \times 0.6^{T/10mins}$$



图二 奖励函数的时间衰减

五、特征与规则

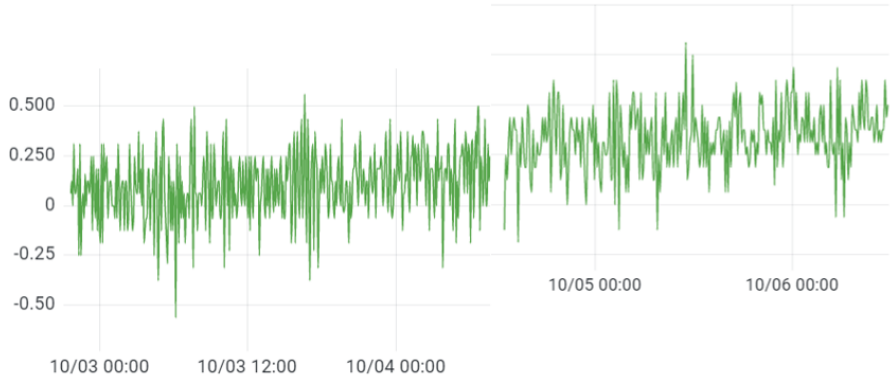
连续的帧中的状态相似度比较高，可以跳过一定量的中间帧而不会丢失太多信息，同时可能可以帮助避免 move 动作抖动卡住。修改 actor 中的相关逻辑（没有验证对不对），使用预训练模型继续训练，但训练效果较差。根据我们对游戏的理解，尝试了不同的召唤师技能，其中马可波罗有尝试过眩晕，但没有对比出明显的优势，所以最终所有英雄都采用狂暴。

六、 模型迭代过程

整个模型的超参数量较多，实验资源也有限，在没有收敛前，很难有一个节省资源且确定的办法，抱着试试看的态度，根据我们对问题的认知进行超参的迭代调整，主要的调整有调高折扣因子，后续实验中调低起始学习率。调低起始学习率是想在局部最优解附近找到更优的解，到了比赛后期我们不希望 agent 跳出我们现在找的一个还不错的解，去寻找其他的局部最优解。调整折扣率会影响梯度下降（上升）寻优的面，因此模型性能先下降后回升，这里也只是一个尝试，我们也无法预估什么样的参数下的局部最优解的性能天花板会更高。

七、 系统工程架构

即使 actor 和 learner 的数量之比是 32:1，但 gamecore 生成样本的速度较慢，远不如网络训练时的样本消耗速度快，导致整个训练过程中的样本消耗比较高，根据开发文档，通过调整 slow_time 参数将样本消耗比例在 8:1 左右，如果资源不受限，能够平衡这种速度差，1:1 能够得到更准确的策略梯度估计，不过 dual-clip ppo 算法本身也允许用于更新的样本与策略网络本身存在一定差异。训练数据是 AI server 通过自博弈生成，可视化面板中的 win-rate 又是默认与内置行为树对战的胜率，模型与模型之间又有着非线性的克制关系，为了更直观的感受我们模型的性能，我们修改了测试相关逻辑，将 common_ai 改为加载我们自己历史效果较好的 ckpt 来评估实施训练效果。



图三 实时模型与历史较好模型对战胜率

八、 训练效果分析

最终模型的训练时长约在 150h（32 actor & 1 learner），各英雄能力水平存在一定差异，与最初测试的一样，英雄之间本就存在克制关系（鲁班>后羿、狄仁杰>公孙离>马可波罗），这个基本规律没有改变，也是符合想像的，1v1 对战中如果没有很好的操作，站桩射手比位移射手有优势的。通过对 reward 的调整，我们成功让英雄改掉了回城回血至 80%路过塔下吃血包的操作。模型的作战风格是大胆的，特别是公孙离的一些操作十分具有观赏性，能够利用好四段位移和格挡，令我们小队成员叹为观止。我们最终的模型惜败 baseline3（星耀）。

九、 总结与展望

正如强化学习中探索与利用的平衡，我们在有限的资源内做了一些探索，未能取得性能提升，所以后期资源都用在训练上，以寻找更加靠近已知局部最优点的更优解。由于时间有限，没有成功找到能取得更好表现的奖励函数，时间和资源允许情况下，可在同样的断点上用同样的实验设置，对比不同 reward 参数的性能，以确定更好的探索方向。如果有机会，希望可以尝试加入图像特征、实现 KDA、发育、伤害、推进相关的多头价值函数的网络[2]。

参考文献

[1] Ye D, Liu Z, Sun M, et al. Mastering complex control in moba games with deep reinforcement learning[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2020, 34(04): 6672-6679.

[2] Ye D, Chen G, Zhang W, et al. Towards playing full moba games with deep reinforcement learning[J]. Advances in Neural Information Processing Systems, 2020, 33: 621-632.