人工智能实践
Artificial Intelligence Practice

DCS3015 Autumn 2022

Chao Yu （余超）

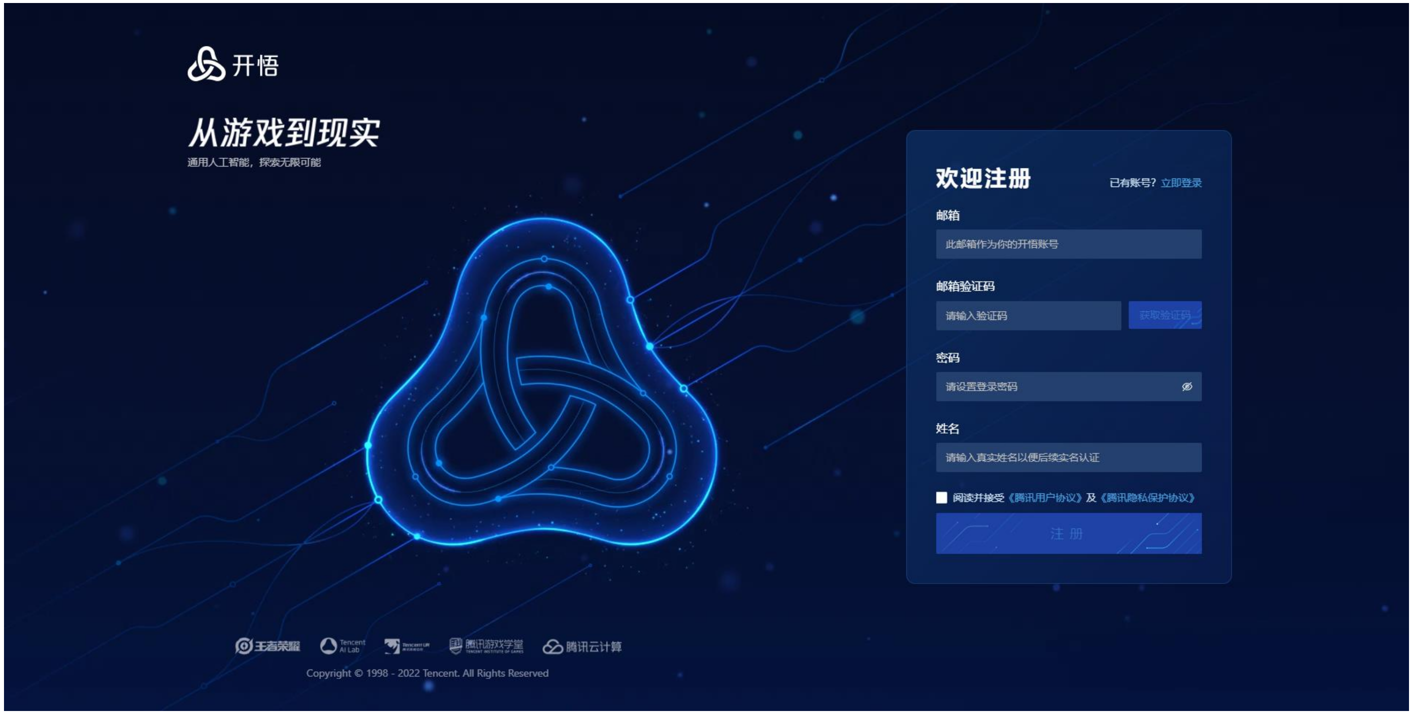
School of Computer Science and Engineering
Sun Yat-Sen University

Lecture：开悟平台介绍

开悟平台介绍

开悟平台的使用

1. 开悟官网注册账号



开悟

从游戏到现实

通用人工智能，探索无限可能

欢迎注册

已有账号？立即登录

邮箱

此邮箱作为你的开悟账号

邮箱验证码

请输入验证码

获取验证码

密码

请设置登录密码

姓名

请输入真实姓名以便后续实名认证

□ 阅读并接受《腾讯用户协议》及《腾讯隐私保护协议》

注 册

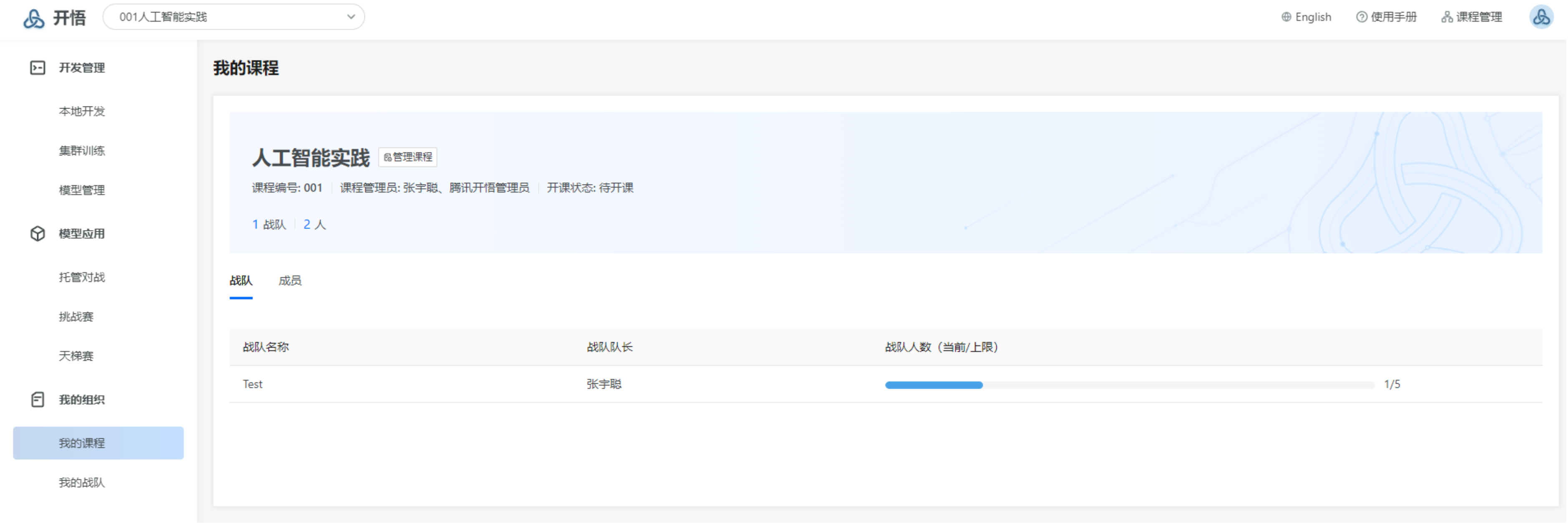
Copyright © 1998 - 2022 Tencent. All Rights Reserved

开悟平台介绍

开悟平台的使用

2. 根据邀请链接加入课程

3. 每五个人一个战队（自行分组）



我的课程

人工智能实践

课程编号: 001 | 课程管理员: 张宇聪、腾讯开悟管理员 | 开课状态: 待开课

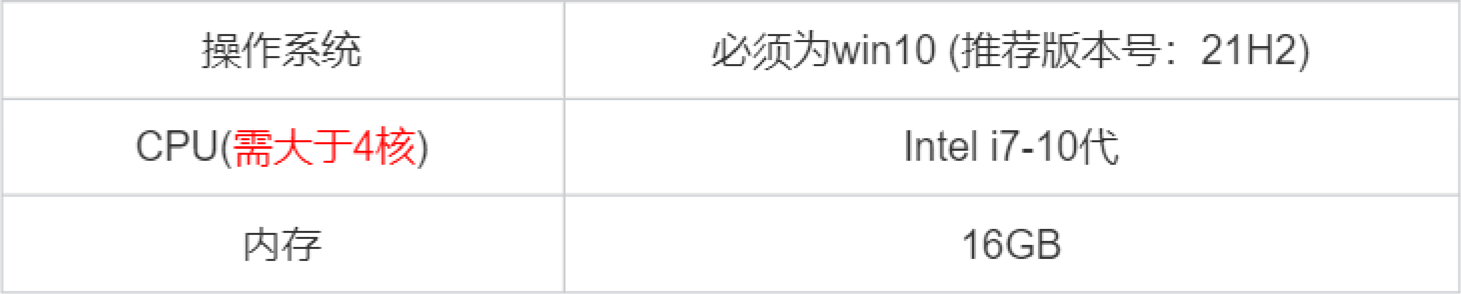
1 战队 | 2 人

战队名称	战队队长	战队人数（当前/上限）
Test	张宇聪	1/5

开悟平台的使用

4. 使用平台前需下载VS Code、docker、开悟客户端（后面会发使用说明文档）

5. 设备要求：

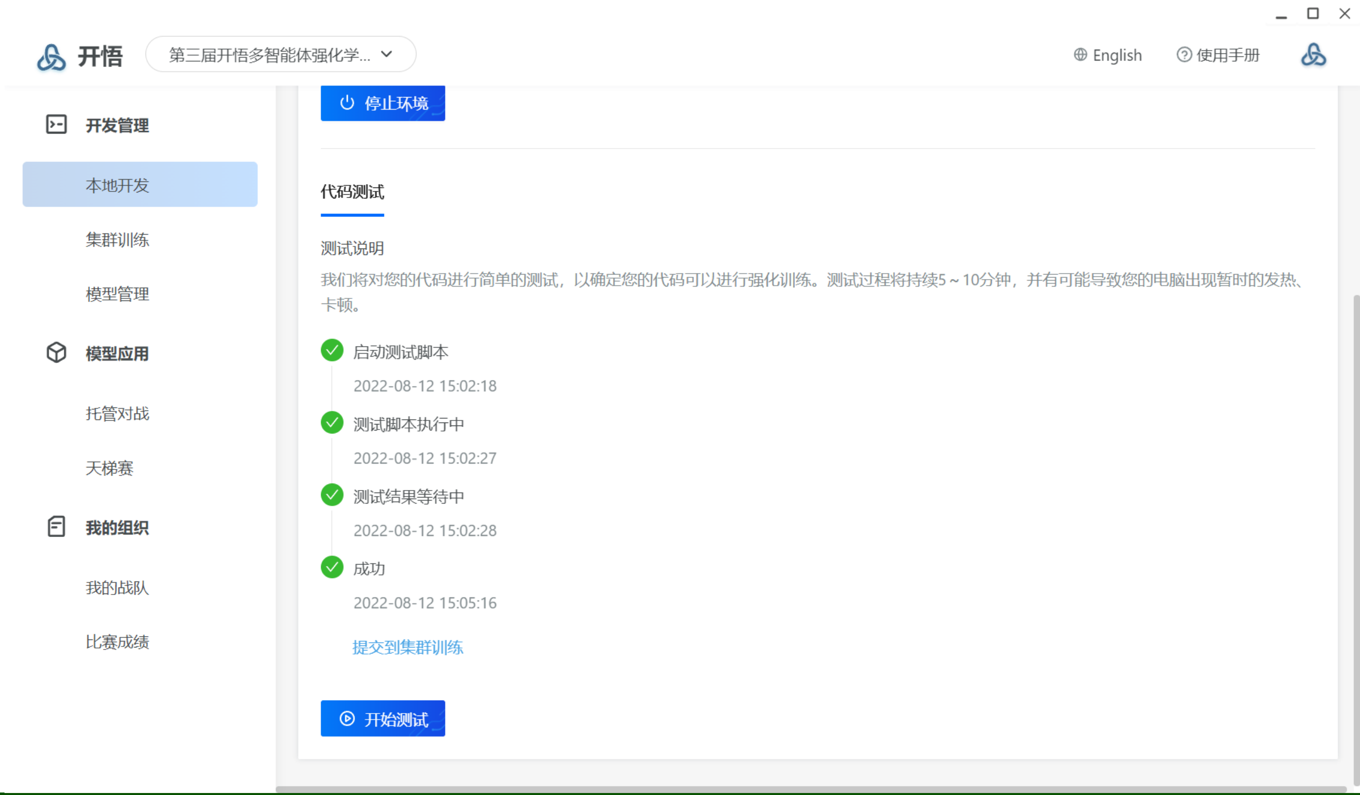


操作系统	必须为win10 (推荐版本号：21H2)
CPU(需大于4核)	Intel i7-10代
内存	16GB

开悟平台介绍

开悟平台的使用

6. 集群训练
   - 本地开发
   - 代码测试
   - 提交集群训练
   - 模型管理



测试说明

我们将对您的代码进行简单的测试，以确定您的代码可以进行强化训练。测试过程将持续5～10分钟，并有可能导致您的电脑出现暂时的发热、卡顿。

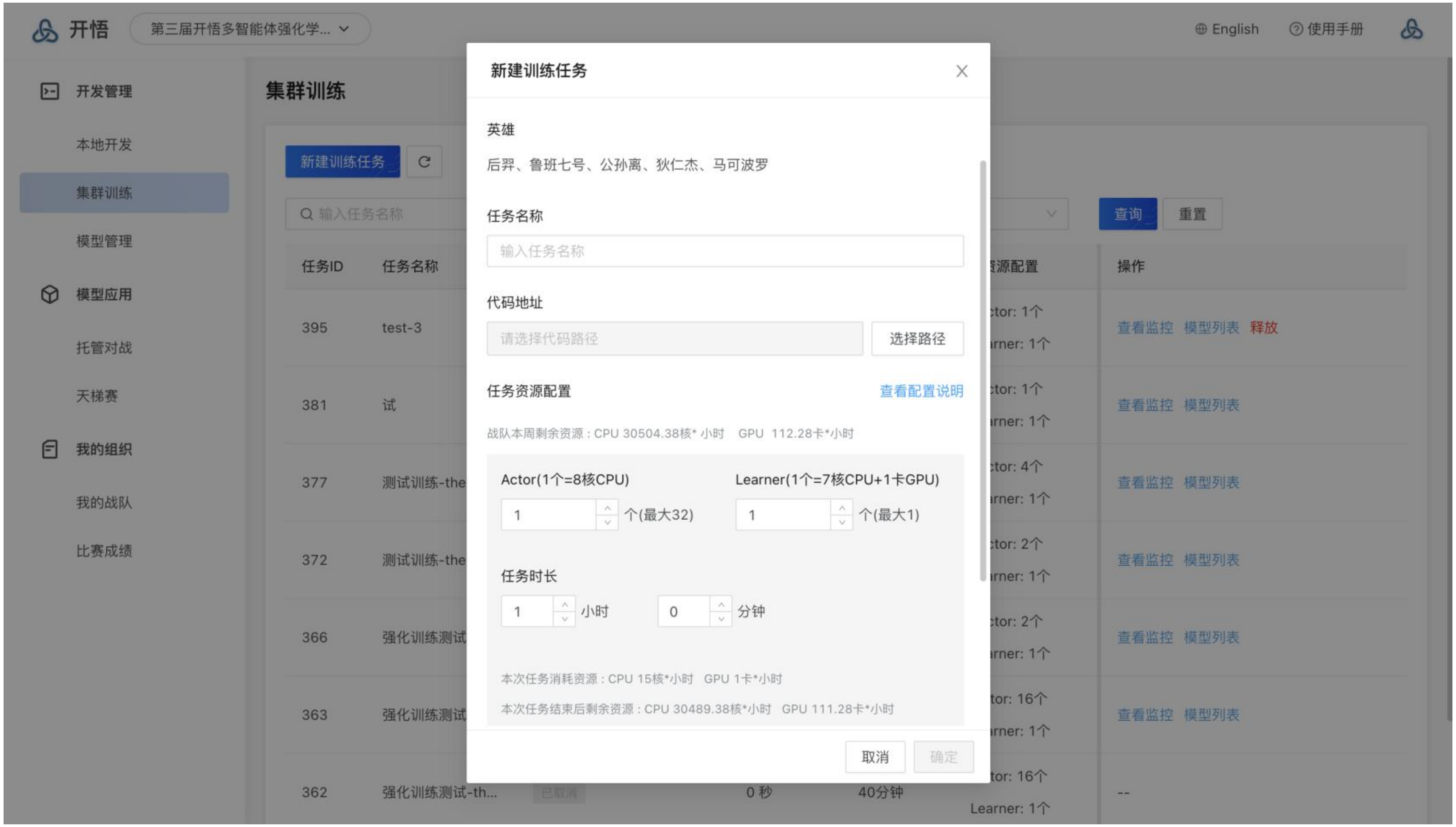
- ✓ 启动测试脚本 2022-08-12 15:02:18
- ✓ 测试脚本执行中 2022-08-12 15:02:27
- ✓ 测试结果等待中 2022-08-12 15:02:28
- ✓ 成功 2022-08-12 15:05:16

提交到集群训练

开悟平台介绍

开悟平台的使用

6. 集群训练



新建训练任务

英雄

后羿、鲁班七号、公孙离、狄仁杰、马可波罗

任务名称

输入任务名称

代码地址

请选择代码路径 选择路径

任务资源配置 查看配置说明

战队本周剩余资源：CPU 30504.38核* 小时 GPU 112.28卡*小时

Actor(1个=8核CPU) 1 个(最大32)

Learner(1个=7核CPU+1卡GPU) 1 个(最大1)

任务时长

1 小时 0 分钟

本次任务消耗资源：CPU 15核*小时 GPU 1卡*小时

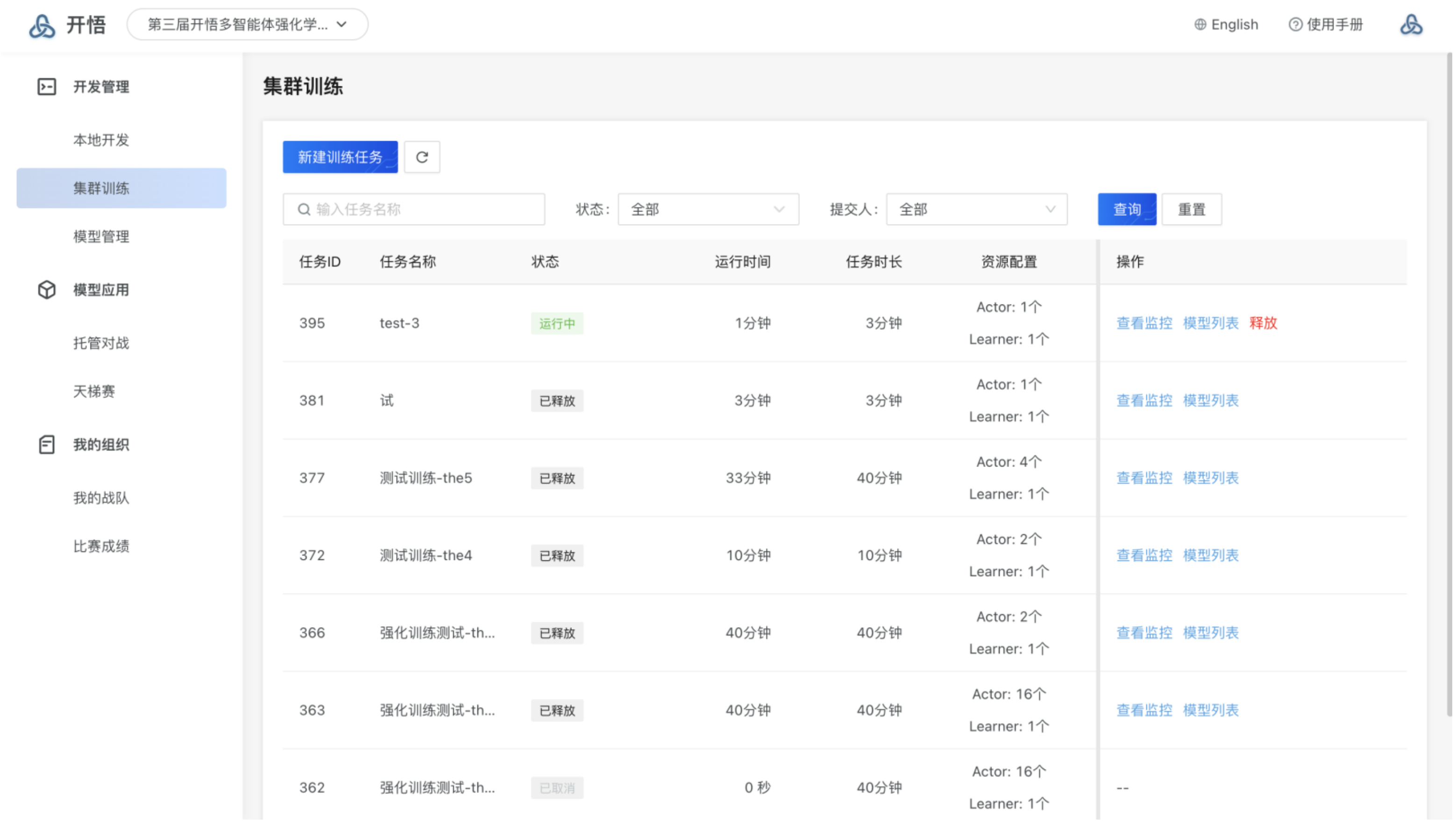
本次任务结束后剩余资源：CPU 30489.38核*小时 GPU 111.28卡*小时

取消 确定

任务ID	任务名称				源配置	操作
395	test-3				tor: 1个 rner: 1个	查看监控 模型列表 释放
381	试				tor: 1个 rner: 1个	查看监控 模型列表
377	测试训练-the				tor: 4个 rner: 1个	查看监控 模型列表
372	测试训练-the				tor: 2个 rner: 1个	查看监控 模型列表
366	强化训练测试				tor: 2个 rner: 1个	查看监控 模型列表
363	强化训练测试				tor: 16个 rner: 1个	查看监控 模型列表
362	强化训练测试-th...	已取消	0 秒	40分钟	tor: 16个 Learner: 1个	--

开悟平台的使用

6. 集群训练



集群训练

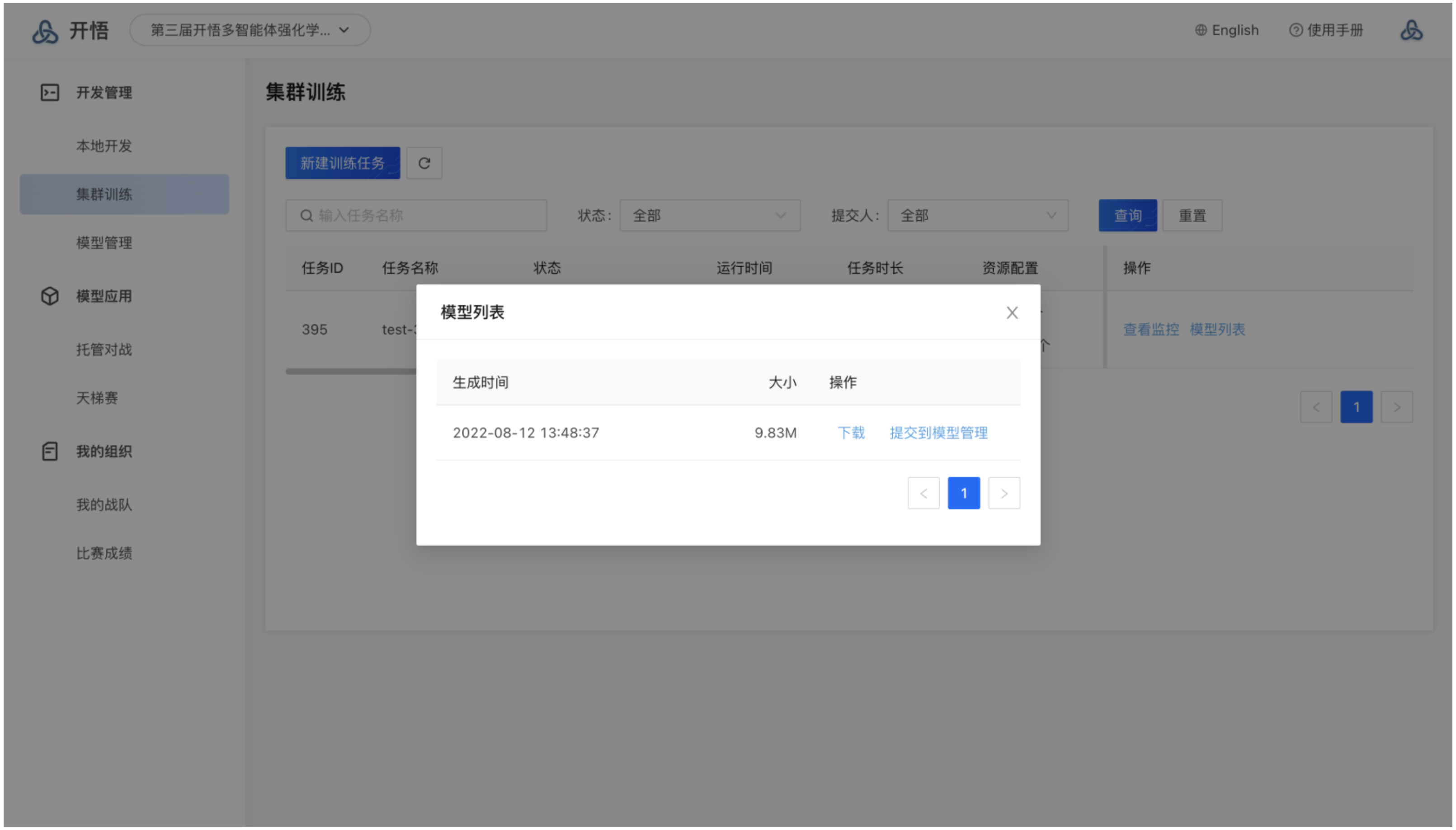
输入任务名称　状态：全部　提交人：全部

任务ID	任务名称	状态	运行时间	任务时长	资源配置	操作
395	test-3	运行中	1分钟	3分钟	Actor: 1个 Learner: 1个	查看监控 模型列表 释放
381	试	已释放	3分钟	3分钟	Actor: 1个 Learner: 1个	查看监控 模型列表
377	测试训练-the5	已释放	33分钟	40分钟	Actor: 4个 Learner: 1个	查看监控 模型列表
372	测试训练-the4	已释放	10分钟	10分钟	Actor: 2个 Learner: 1个	查看监控 模型列表
366	强化训练测试-th...	已释放	40分钟	40分钟	Actor: 2个 Learner: 1个	查看监控 模型列表
363	强化训练测试-th...	已释放	40分钟	40分钟	Actor: 16个 Learner: 1个	查看监控 模型列表
362	强化训练测试-th...	已取消	0 秒	40分钟	Actor: 16个 Learner: 1个	--

开悟平台介绍

开悟平台的使用

6. 集群训练



任务ID	任务名称	状态	运行时间	任务时长	资源配置	操作
395	test-					查看监控 模型列表

模型列表

生成时间	大小	操作
2022-08-12 13:48:37	9.83M	下载 提交到模型管理

开悟平台介绍

开悟平台的使用

6. 开悟仅能使用TensorFlow深度学习框架（自行学习）

7. 开悟平台提供三个功能模块检验模型能力
   - 托管对战：用户可创建对战任务，使用自己的模型与课程中战队/管理员共享的模型进行对战。同样，也可以选择自己的不同模型进行对战。
   - 挑战赛：在挑战赛中，课程成员的模型可与课程管理员指定的课程AI模型进行对战，并在比赛结束后获得成绩。
   - 天梯赛：不同战队提交的模型会按照比赛设置进行循环对战，即每个战队提交的模型均会与其他所有战队提交的模型进行对战。比赛后会根据比赛胜场展示战队排名结果。

开悟平台介绍

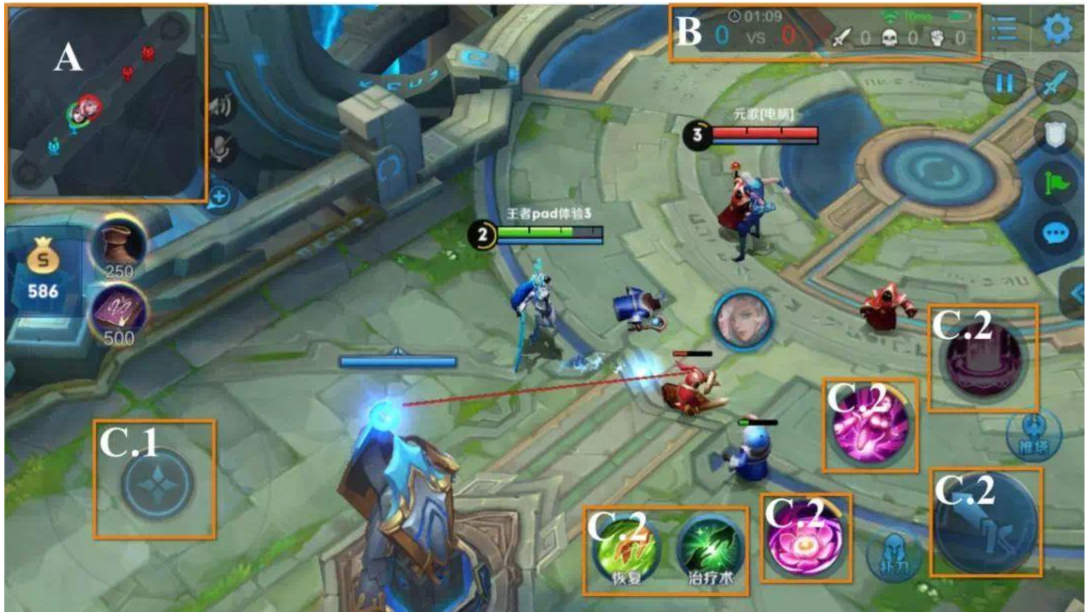
智能体介绍

游戏单元：

- 英雄：友方英雄和敌方英雄
  - 英雄基础属性，包含血量、蓝量、攻击力，防御力，抗性等
  - 英雄技能，包含技能槽、技能CD，技能等级
- 小兵
  - 包含小兵血量、位置等
- 塔
  - 包含塔的位置、血量等特征

Action Space设计：

- 移动键 Move（C1）
- 技能槽(C2)
  - 1技能 Skill1（方向型）
  - 2技能 Skill2（方向型）
  - 3技能 Skill3（自身目标释放，例如貂蝉）
  - 普通攻击 ComAttack（目标型）

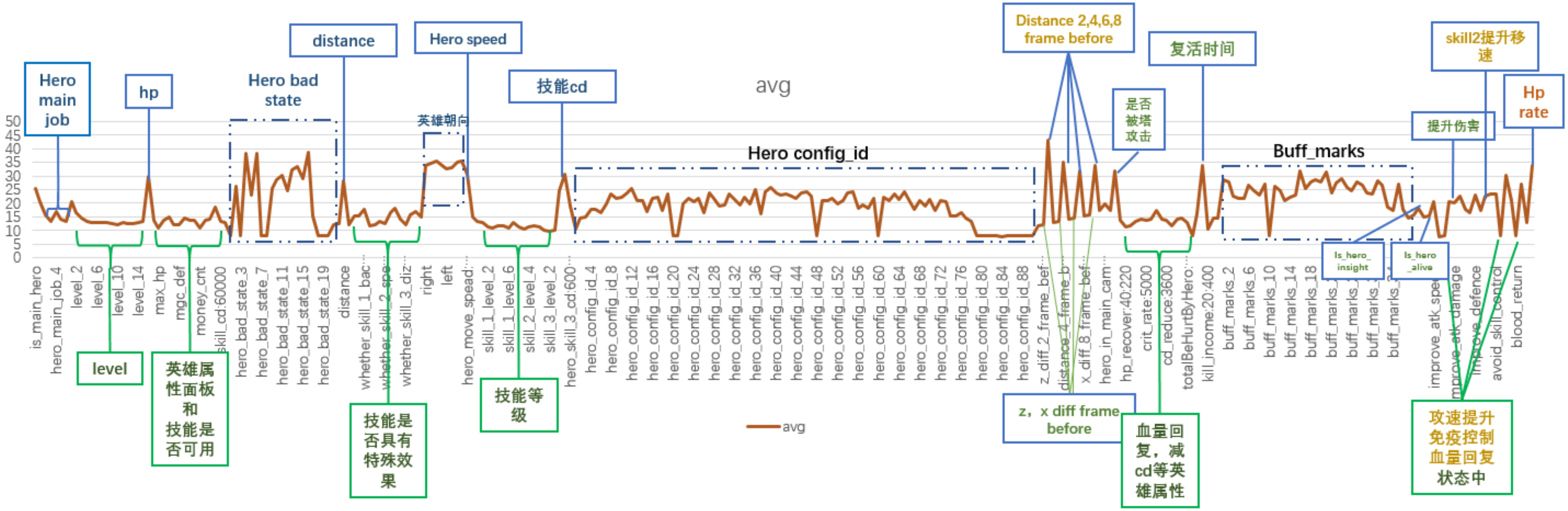


开悟平台介绍

智能体介绍

智能体观测：主英雄特征、友方、敌方、己方小兵、敌方小兵、己方塔、敌方塔、全局信息。

主英雄特征：

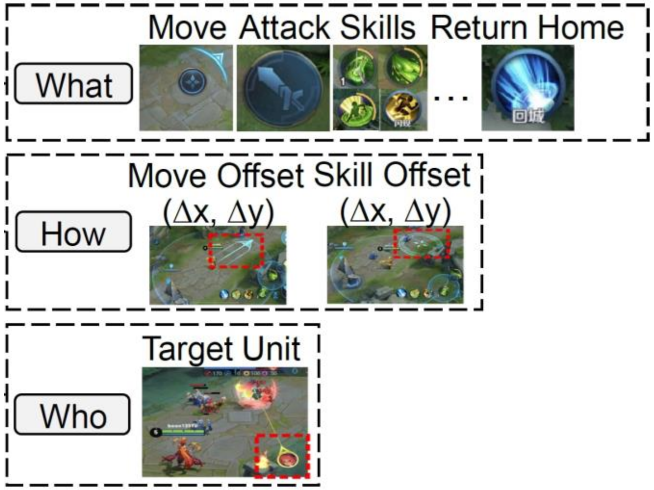


开悟平台介绍

智能体介绍

Action Space设计：

- 移动键 Move（C1）
- 技能槽(C2)
  - 1技能 Skill1（方向型）
  - 2技能 Skill2（方向型）
  - 3技能 Skill3（自身目标释放，例如貂蝉）
  - 普通攻击 ComAttack（目标型）

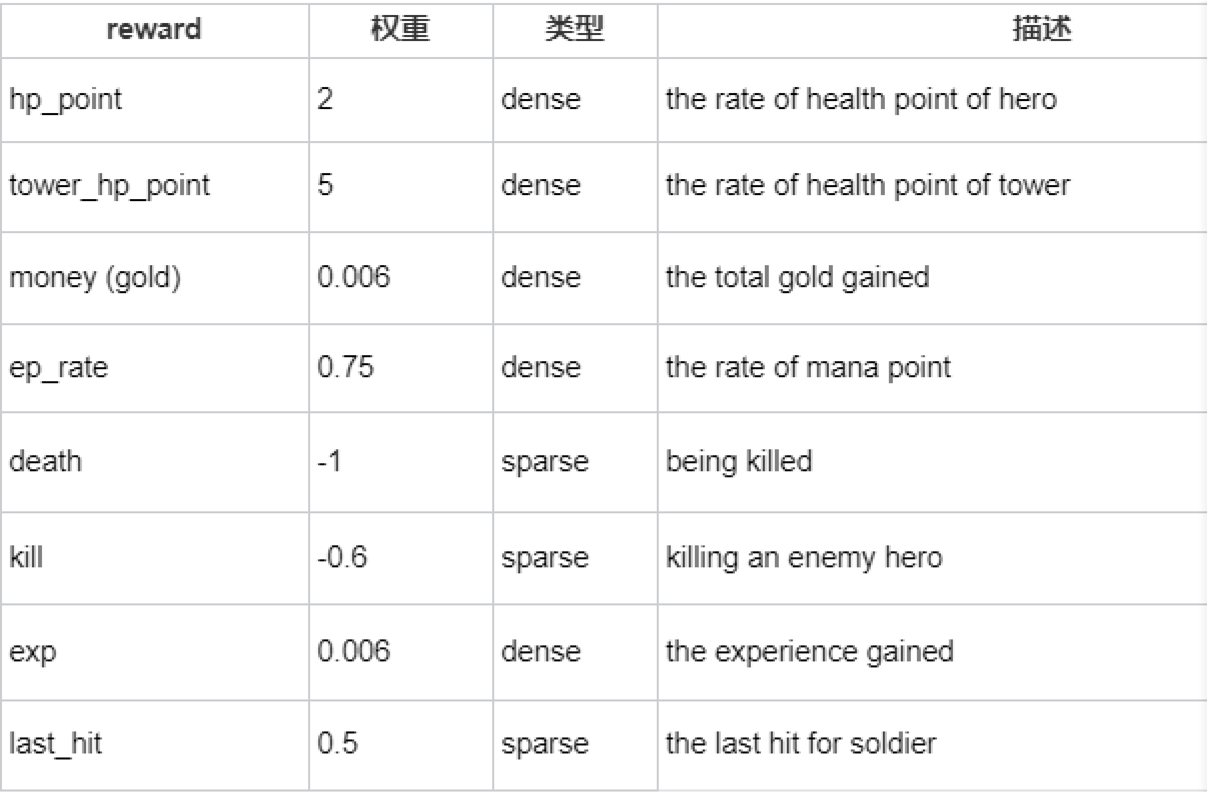


- 游戏中的动作设计
  - what，你要按哪个按键：12个button
  - how，你要往哪个方向拖动按键：16*16个方向选择
  - who，你的技能作用对象是谁：8个target（两塔，四兵，一个英雄，以及None)

开悟平台介绍

智能体介绍

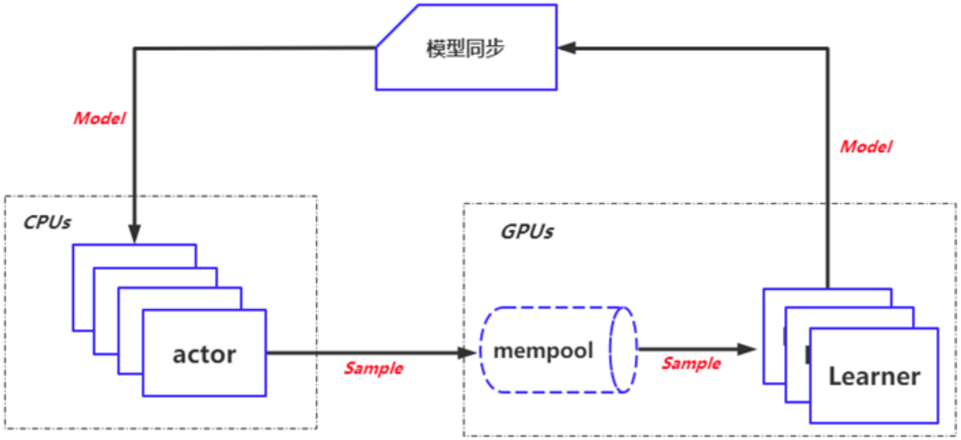
奖励值设计：

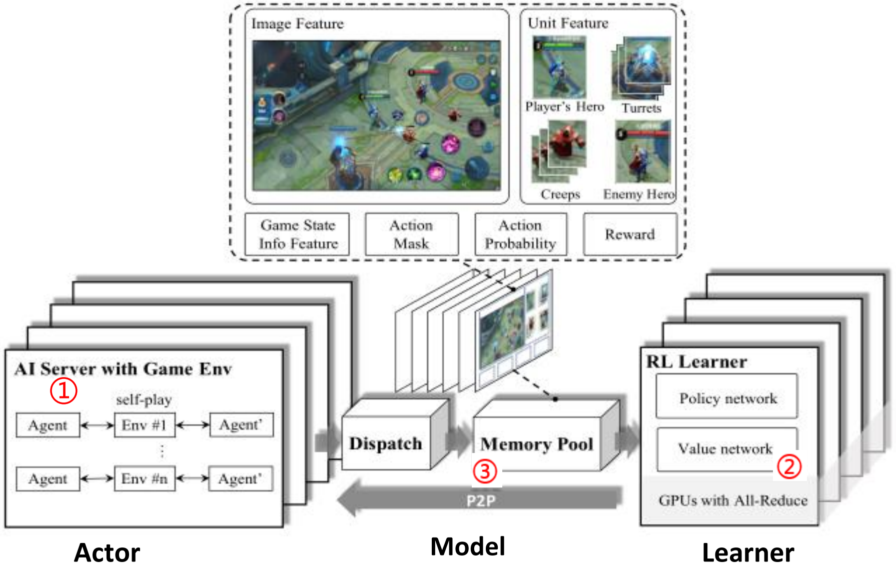


reward	权重	类型	描述
hp_point	2	dense	the rate of health point of hero
tower_hp_point	5	dense	the rate of health point of tower
money (gold)	0.006	dense	the total gold gained
ep_rate	0.75	dense	the rate of mana point
death	-1	sparse	being killed
kill	-0.6	sparse	killing an enemy hero
exp	0.006	dense	the experience gained
last_hit	0.5	sparse	the last hit for soldier

开悟平台介绍

集群训练





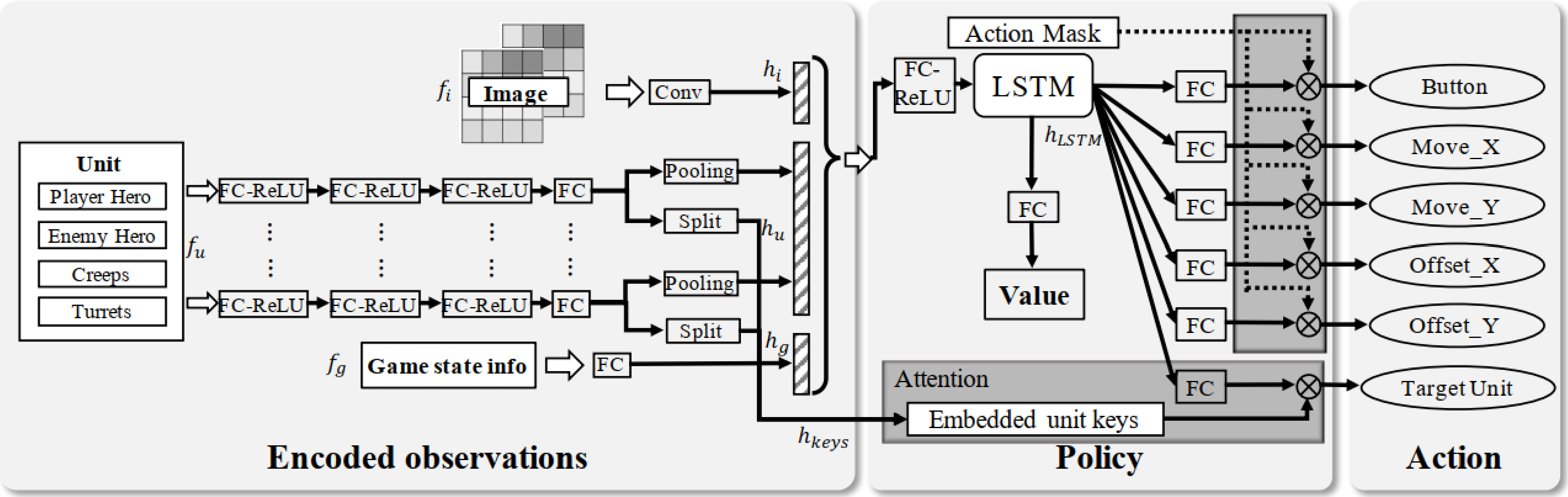
1 自博弈生产数据

2 训练消费数据

3 生成模型 模型同步

开悟平台介绍

官方代码



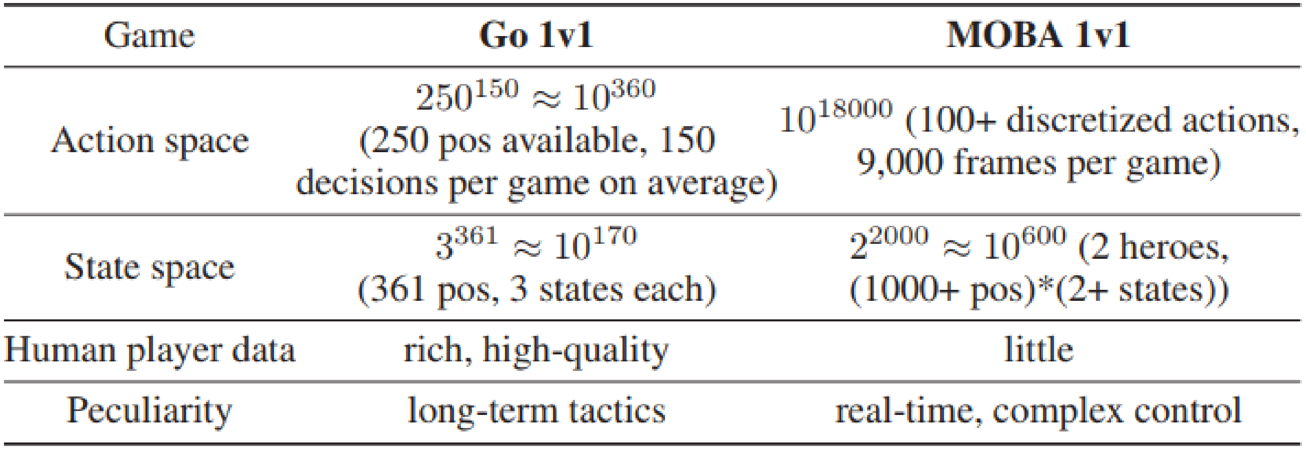
- 单智能体PPO算法
- Multi-head Policy
- Image输入和向量输入
- Attention
- 参数共享
- 特征工程

Ye, Deheng, et al. "Mastering complex control in moba games with deep reinforcement learning." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 2020.

论文介绍（Introduce）

- It’s very difficult to search any policies with human-level performance in the Multi-player Online Battle Arena (MOBA) 1v1 games.
- MOBA 1v1 is a real-time strategy (RTS) game that requires highly complex action control.

Table 1: Comparing Go and MOBA 1v1



Game	**Go 1v1**	**MOBA 1v1**
Action space	$250^{150} \approx 10^{360}$ (250 pos available, 150 decisions per game on average)	10^{18000} (100+ discretized actions, 9,000 frames per game)
State space	$3^{361} \approx 10^{170}$ (361 pos, 3 states each)	$2^{2000} \approx 10^{600}$ (2 heroes, (1000+ pos)*(2+ states))
Human player data	rich, high-quality	little
Peculiarity	long-term tactics	real-time, complex control

Ye, Deheng, et al. "Mastering complex control in moba games with deep reinforcement learning." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 2020.

论文介绍（introduce）

- The complexity of MOBA 1v1 also comes from the playing mechanism：
  - To win a game, in the partially observable environment, agents must learn to plan, attack, defend, control skill combos, induce, and deceive the opponents.
  - Apart from the player’s and the opponent’s agent, there exists many more game units, e.g., creeps and turrets. This creates challenge to the target selection which requires delicate sequences of decision making and corresponding action controls.
  - Different heroes in a MOBA game have very different playing methods. The action control can completely change from hero to hero, which calls for robust and unified modeling.

Ye, Deheng, et al. "Mastering complex control in moba games with deep reinforcement learning." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 2020.

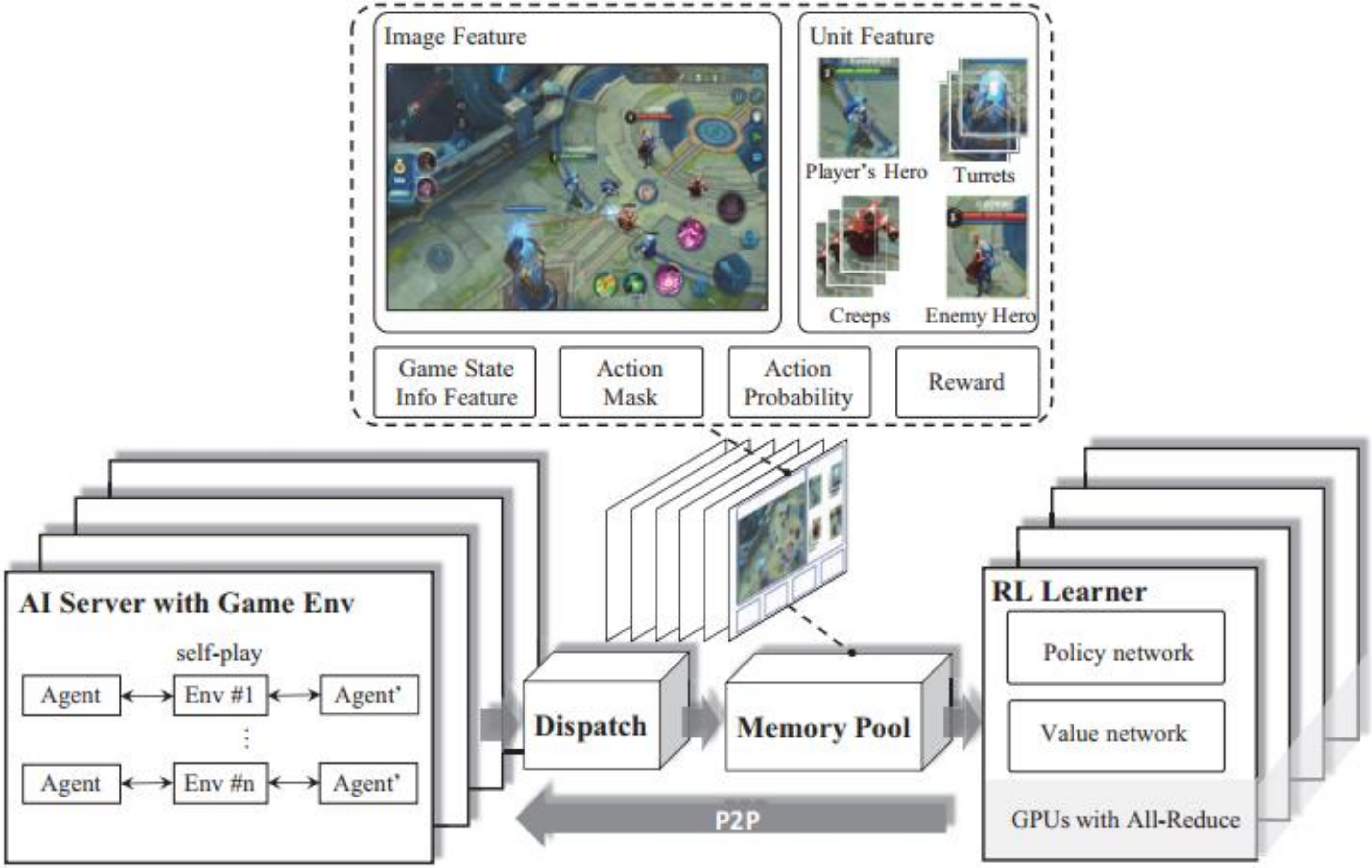
论文介绍（Method）

- Considering the fact that complex agent control problems can introduce high variance of stochastic gradients, e.g. the MOBA 1v1 games, large batch size is necessary to speed up the training
- Designing a scalable and loosely-coupled system architecture to construct the utility of data parallelism：
    - **Reinforcement Learning (RL) Learner**
    - **Artificial Intelligence (AI) Server**
    - **Dispatch Module**
    - **Memory Pool**

Ye, Deheng, et al. "Mastering complex control in moba games with deep reinforcement learning." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 2020.

开悟平台介绍

论文介绍（Method）



Ye, Deheng, et al. "Mastering complex control in moba games with deep reinforcement learning." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 2020.

论文介绍（Method）

- **Artificial Intelligence (AI) Server：**
  - The AI Server implements how the AI model interacts with the environment. AI server covers the interaction logic between game environment and the AI model , and generates episodes via self-play with mirrored policies.
- **Dispatch Module**
  - The Dispatch Module is a station for sample collection, compression and transmission. It is a server that collects data samples from AI Servers, consisting of reward, feature, action probabilities, etc.
- **Memory Pool**
  - The Memory Pool is the data storage module, which provides training instances for the RL Learner. It supports samples of varied lengths, and data sampling based on the generated time.
- **Reinforcement Learning (RL) Learner**
  - The **RL Learner** is a distributed training environment. The gradients in the RL learners are averaged through the ring allreduce algorithm.

Ye, Deheng, et al. "Mastering complex control in moba games with deep reinforcement learning." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 2020.

论文介绍（Method）

- **Algorithm Design (Novel Strategies)：**
  - The *target attention mechanism* is designed in this network to help with the target selection in MOBA combats.
  - LSTMs are leveraged for the hero to learn the skill combos which are critical to create severe and instant damage
  - The decoupling of control dependencies is conducted to form a multi-label proximal policy optimization (PPO) objective
  - A game-knowledge-based pruning method, called *action mask*, is developed to guide explorations during the reinforcement process
  - A dual-clipped version of the PPO algorithm is proposed to guarantee convergence with large and deviated batches

Ye, Deheng, et al. "Mastering complex control in moba games with deep reinforcement learning." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 2020.

开悟平台介绍

论文介绍（Method）

- **Algorithm Design：**
  - In the RL Learner, an actor-critic network is implemented to model the action control dependencies in MOBA 1v1 games.

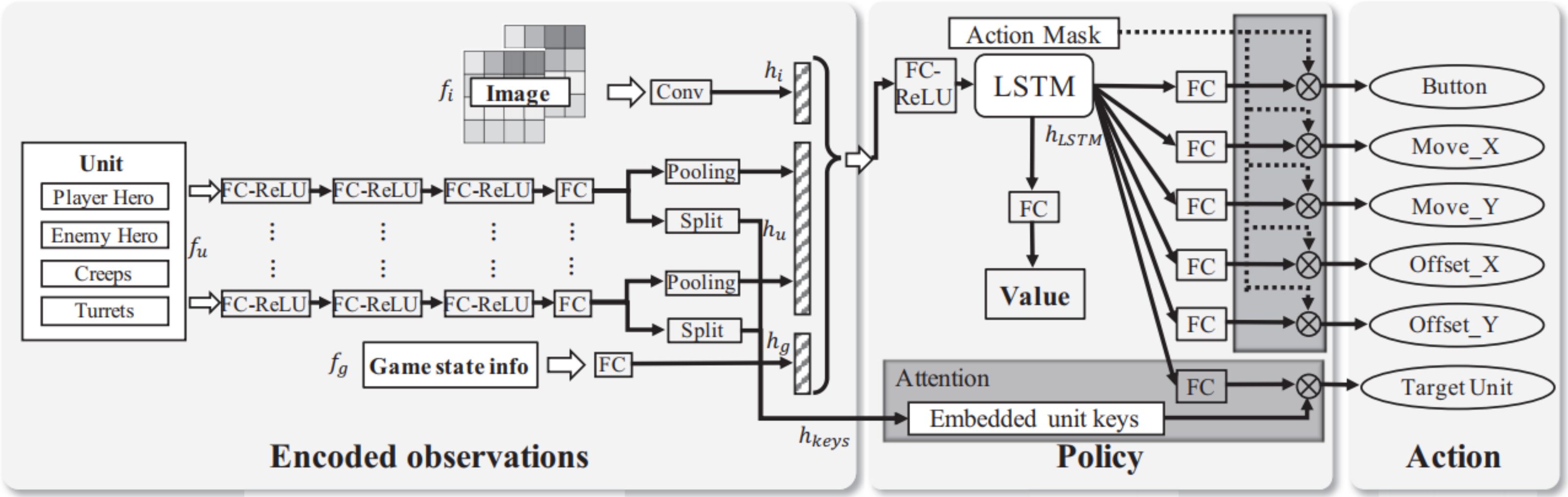


Figure 2: Illustrations of state, action, policy and value. A state $s \in \mathcal{S}$ covers three types of information: local image info (e.g., obstacles in 2D), observable unit attributes (e.g., hero type, health point), and observable game state info (e.g., game time, turrets destroyed, etc.), i.e., $s = [f_i, f_u, f_g]$. An action $a \in \mathcal{A}$ in a MOBA 1v1 game specifies two items: the content (i.e., the action button to press, the Move_X, the Move_Y, Offset_X, and Offset_Y) and the target game unit. The action buttons include move, attack, skill releasing, etc. The policy π_θ is modeled by FCs and an LSTM, which also predicts the values.

Ye, Deheng, et al. "Mastering complex control in moba games with deep reinforcement learning." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 2020.

论文介绍（Method）

- **Algorithm Design (Novel Strategies)：**
  - The network encodes image features f_i, vector features f_u, and game state information f_g (the observable game states) as encodings h_i, h_u, and h_g using convolutions, fully-connections, and fully-connections (FC), respectively.
  - To handle the varied number of units, the same type units are mapped to a feature vector of fixed length by max-pooling
  - The state encoding is then mapped to the final representation h_{LSTM} by a LSTM cell, which further takes the temporal information into consideration.
  - The target unit *t* of action *a* is predicted by a **target attention** mechanism over every unit. This mechanism treats a FC output of h_{LSTM} as the query, the stack of all unit encodings as the keys h_{KEYS}, and calculate the target attention as:

$$p(t|a) = Softmax(FC(h_{LSTM}) \cdot h_{keys}^{T})$$

Ye, Deheng, et al. "Mastering complex control in moba games with deep reinforcement learning." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 2020.

论文介绍（Method）

- **Algorithm Design (Novel Strategies)：**
  - It is very hard to explicitly model the intercorrelations among different labels in one action of MOBA games in the multi-label policy network
  - Treat each label in an action independently to decouple their intercorrelations, i.e., **the decoupling of control dependencies:**

$$\max_{\theta} \hat{\mathbb{E}}_{s,a\sim\pi_{\theta_{\text{old}}}}\left[\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}\hat{A}_t\right], \quad \max_{\theta} \sum_{i=0}^{N_a-1} \hat{\mathbb{E}}_{s,a\sim\pi_{\theta_{\text{old}}}}\left[\frac{\pi_\theta(a_t^{(i)}|s_t)}{\pi_{\theta_{\text{old}}}(a_t^{(i)}|s_t)}\hat{A}_t\right]$$

  - **Dual-clip PPO**
    - Because the ratio $r_t(\theta)$ can be extremely large, maximization of the RL objective may lead to an excessively large policy deviation. To alleviate this issue, the standard PPO algorithm involves a ratio clip

$$\mathcal{L}^{\text{CLIP}}(\theta) = \hat{\mathbb{E}}_t\Big[\min\Big(r_t(\theta)\hat{A}_t, \text{clip}\big(r_t(\theta), 1-\epsilon, 1+\epsilon\big)\hat{A}_t\Big)\Big],$$

Ye, Deheng, et al. "Mastering complex control in moba games with deep reinforcement learning." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 2020.

论文介绍（Method）

- **Algorithm Design (Novel Strategies)：**
  - **Dual-clip PPO**
    - However, in large-scale off-policy training environments, the trajectories are sampled from various sources of policies, which may differ considerably from the current policy π_θ. In such situations, the standard PPO will fail to work with such deviations since it was originally proposed for on-policy: for example, when $\pi_\theta\left(a_t^{(i)} \middle| s_t\right) \gg \pi_{\theta_{old}}\left(a_t^{(i)} \middle| s_t\right)$, the ratio $r_t(\theta)$ is a huge number. When $\hat{A}_t < 0$, such a large ratio $r_t(\theta)$ will introduce a big and unbounded variance since $r_t(\theta)\hat{A}_t \ll 0$.

    - Proposing a dual-clipped PPO algorithm to support large-scale distributed training, which further clip the ratio $r_t(\theta)$ with a lower bound of the value $r_t(\theta)\hat{A}_t$. When $\hat{A}_t < 0$, the new objective of our dual-clipped PPO is

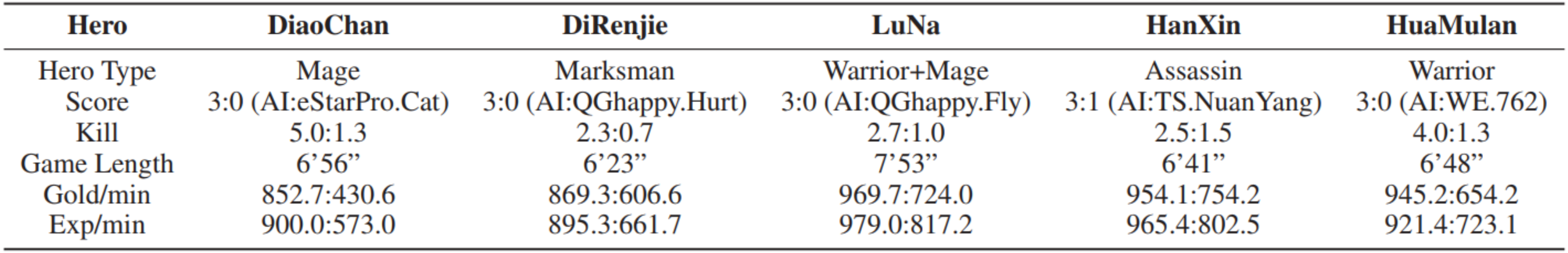
$$\hat{\mathbb{E}}_t\left[\max\left(\min\left(r_t(\theta)\hat{A}_t, \text{clip}\left(r_t(\theta), 1-\epsilon, 1+\epsilon\right)\hat{A}_t\right), c\hat{A}_t\right)\right]$$

Ye, Deheng, et al. "Mastering complex control in moba games with deep reinforcement learning." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 2020.

论文介绍（Method）

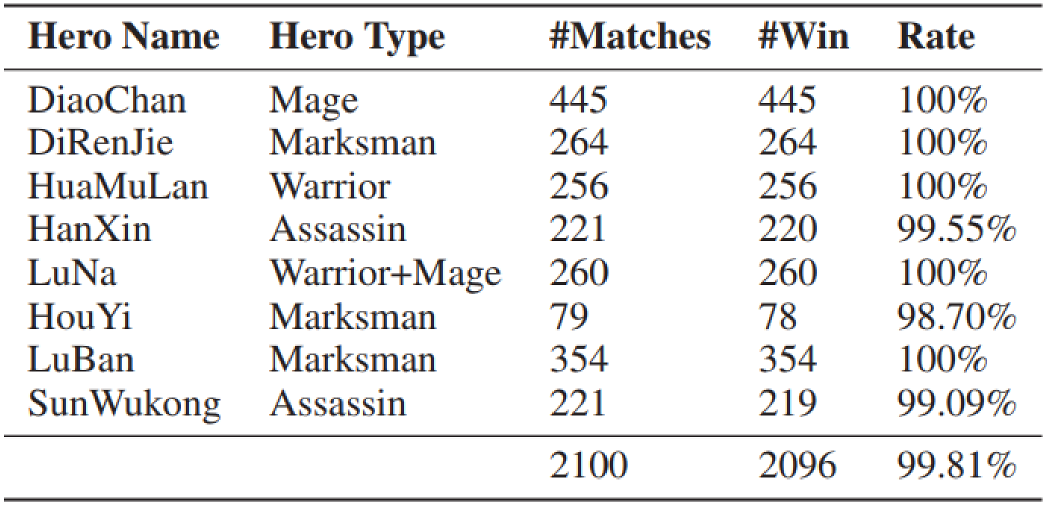
- **Algorithm Design (Results)：**

Table 3: Match Statistics of our AI vs. Professional Players on Different Types of Heroes



Hero	**DiaoChan**	**DiRenjie**	**LuNa**	**HanXin**	**HuaMulan**
Hero Type	Mage	Marksman	Warrior+Mage	Assassin	Warrior
Score	3:0 (AI:eStarPro.Cat)	3:0 (AI:QGhappy.Hurt)	3:0 (AI:QGhappy.Fly)	3:1 (AI:TS.NuanYang)	3:0 (AI:WE.762)
Kill	5.0:1.3	2.3:0.7	2.7:1.0	2.5:1.5	4.0:1.3
Game Length	6’56”	6’23”	7’53”	6’41”	6’48”
Gold/min	852.7:430.6	869.3:606.6	969.7:724.0	954.1:754.2	945.2:654.2
Exp/min	900.0:573.0	895.3:661.7	979.0:817.2	965.4:802.5	921.4:723.1

Table 4: Results of AI vs. Various Top Human Players



Hero Name	**Hero Type**	**#Matches**	**#Win**	**Rate**
DiaoChan	Mage	445	445	100%
DiRenJie	Marksman	264	264	100%
HuaMuLan	Warrior	256	256	100%
HanXin	Assassin	221	220	99.55%
LuNa	Warrior+Mage	260	260	100%
HouYi	Marksman	79	78	98.70%
LuBan	Marksman	354	354	100%
SunWukong	Assassin	221	219	99.09%
		2100	2096	99.81%

Ye, Deheng, et al. "Mastering complex control in moba games with deep reinforcement learning." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 2020.

Thanks