



How To ML-Agents

: 예제로 알아보기

UPnL 정회원 류지민

UPnL 여름 워크샵



발표자 소개



Jimin Ryu

nick11967

- 게임 개발 경력 6개월
- <골든링> 프로젝트 진행 중
- 팀에서 프로그래머, PM, 류지민을 담당 중
- 최근에 ml-agents를 알게 됨
- 양자 정보 & 컴퓨팅에 관심 있음

골드링 소개



- UNIJAM에서 시작
- UPnL에서는 저 혼자, G-Pos, KING과 함께
- Team AMUZ
- 디지털 전략 보드게임
- 자원을 이용해 카드를 사면 그 카드가 다시 자원을 생산하고, 그렇게 불린 지원으로 다시 카드를 사며 자원을 늘려가는 게임
- 2023 인디크래프트 챌린저 부문 Top 10
- 최근 알파 v0.1 유틸리티 한정 출시

Why ml-agents

Unity

많은 자료

ML 지식 X

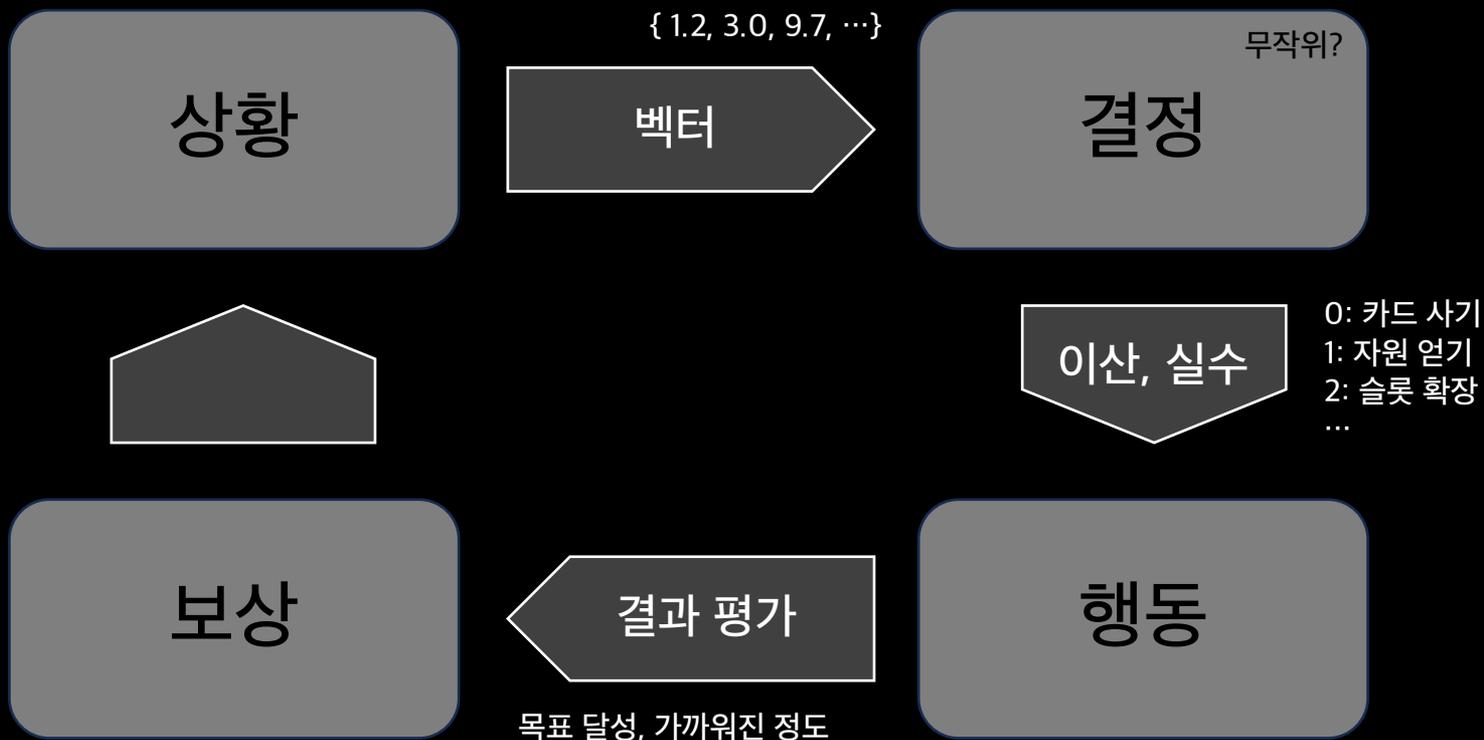
What

4인 플레이

시스템의 변경

카드 밸런스
테스트

How - 강화학습 이미지 얻기



How - 유니티 스크립트 작성

OnEpisodeBegin()

새로운 에피소드 시작, 환경 초기화

CollectObservations(VectorSensor sensor)

강화학습 프로그램에게 관측 정보 전달

```
sensor.AddObservation(정보);
```

정보의 Type: bool, int, float, IList<float>

OnActionReceived(ActionBuffers actionBuffers)

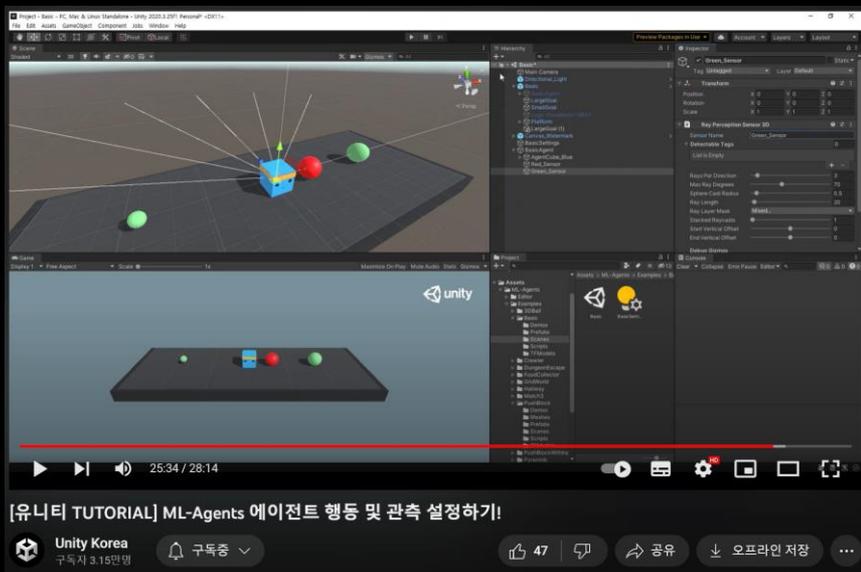
강화학습을 통한 행동이 결정되는 곳

```
var discrete = actionBuffers.DiscreteActions[0];
```

How - 유니티 프로퍼티

좋은 자료들을 살펴봅시다! 링크를 맨 마지막에 모아뒀습니다.

[유튜브 유니티 코리아 튜토리얼]



[누군가 친절히 보여주는 블로그]

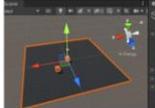
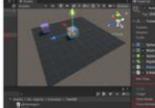
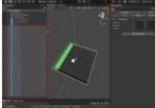
'유니티 > mlAgents' 카테고리의 다른 글

- 유니티 머신러닝 개발 ML Agents 8편, 목표 찾기 예제 개선. 2 floor 확장, 복잡한 바닥형태 테스트와 학습의 한계점 (0) 2021.04.07
- 유니티 머신러닝 개발 ML Agents 7편, 목표 찾기 예제 개선. 1 뷰임변경, 회전제한과 스텝제한, 환경개선과 재교육 (3) 2021.04.07
- 유니티 머신러닝 개발 ML Agents 5편, 목표찾기 예제 제작.1 목표찾기 유니티 프로그램 제작 (6) 2021.04.06
- 유니티 머신러닝 개발 ML Agents 4편, 강화 학습시켜보고, 결과를 이용해 보기 (5) 2021.04.05
- 유니티 머신러닝 개발 ML Agents 3편, ml agents 파이션 패키지 설치하기 (5) 2021.04.04

태그

ML mlagent 강화학습 머신러닝 목표찾기 유니티

관련글

-  유니티 머신러닝 개발 ML ...
-  유니티 머신러닝 개발 ML ...
-  유니티 머신러닝 개발 ML ...
-  유니티 머신러닝 개발 ML ...

How - 하이퍼 파라미터

Write config.yaml

```
##title Write config.yaml

# Code below is an example code
%%writefile config.yaml
behaviors:
  MoveToGoal:
    trainer_type: sac

# Trainer configurations common to all trainers
max_steps: 2800000 # Typical range: 5e5 - 1e7
time_horizon: 64 # Typical range: 32 - 2048
summary_freq: 10000
keep_checkpoints: 5
checkpoint_interval: 100000
threaded: false

hyperparameters:
  # Hyperparameters common to PPO and SAC
  batch_size: 512 # Typical range: (Continuous - PPO): 512 - 5120
  buffer_size: 50000 # Typical range: PPO: 2048 - 409600; SAC: 50000 - 1000000
  learning_rate: 0.0003 # Typical range: 1e-5 - 1e-3
  learning_rate_schedule: constant # linear for PPO or constant for SAC
  # SAC-specific hyperparameters
  buffer_init_steps: 0
  tau: 0.005
  steps_per_update: 20.0
  save_replay_buffer: false
  init_entcoef: 1.0
  reward_signal_steps_per_update: 20.0
# Configuration of the neural network (common to PPO/SAC)
network_settings:
```

How - 학습 진행



+서울대 컴공 GPU

How - 학습 진행

```
buffer_size: 12000
epsilon: 0.2
hidden_units: 128
...
lambda: 0.99
learning_rate: 0.009
learning_rate_scheduler: linear
max_steps: 5.065
memory_size: 128
normalize: True
num_epoch: 3
num_layers: 2
time_horizon: 1000
sequence_length: 64
summary_freq: 12000
use_recurrent: False
vis_encode_type: simple
reward_signals:
  extrinsic: 1.0
  strength: 0.99
  gamma: 0.99
summary_path: test-1_30Ball
model_path: ./models/test-1/30Ball
keep_checkpoints: 5

/content/ml-agents/ml-agents/trainers/models.py:268: UserWarning: `tf.layers.dense` is deprecated and will be removed in a future version. Please use `tf.keras.layers.Dense` instead.
hidden = tf.layers.dense(
/content/ml-agents/ml-agents/trainers/distributions.py:112: UserWarning: `tf.layers.dense` is deprecated and will be removed in a future version. Please use `tf.keras.layers.Dense` instead.
nu = tf.layers.dense(
/content/ml-agents/ml-agents/trainers/models.py:639: UserWarning: `tf.layers.dense` is deprecated and will be removed in a future version. Please use `tf.keras.layers.Dense` instead.
value = tf.layers.dense(hidden_input, 1, name="{}_value".format(name))
2023-07-28 01:35:36.56484: tensorflow/c/c_api.cc:300] Operation '(name:policy/log_std/Assign) id:126 op device:(requested: '', assigned: '') def:{{node policy/log_std/Assign}} = Assign[T=DT_FLOAT, _class=["loc:policy/log_std"], _has_manual_control_
2023-07-28 01:35:49 INFO [stats.py:100] test-1_30Ball | Step: 12000. Time Elapsed: 15.513 s Mean Reward: 1.175. Std of Reward: 0.705. Training.
2023-07-28 01:35:05 INFO [stats.py:100] test-1_30Ball | Step: 24000. Time Elapsed: 31.008 s Mean Reward: 1.300. Std of Reward: 0.781. Training.
2023-07-28 01:35:21 INFO [stats.py:100] test-1_30Ball | Step: 36000. Time Elapsed: 46.784 s Mean Reward: 1.771. Std of Reward: 1.161. Training.
2023-07-28 01:35:36 INFO [stats.py:100] test-1_30Ball | Step: 48000. Time Elapsed: 62.066 s Mean Reward: 2.500. Std of Reward: 1.726. Training.
2023-07-28 01:35:50 INFO [stats.py:100] test-1_30Ball | Step: 60000. Time Elapsed: 76.110 s Mean Reward: 4.000. Std of Reward: 3.193. Training.
2023-07-28 01:37:04 INFO [stats.py:100] test-1_30Ball | Step: 72000. Time Elapsed: 89.715 s Mean Reward: 7.557. Std of Reward: 7.403. Training.
2023-07-28 01:37:15 INFO [stats.py:100] test-1_30Ball | Step: 84000. Time Elapsed: 101.577 s Mean Reward: 20.590. Std of Reward: 19.444. Training.
2023-07-28 01:37:29 INFO [stats.py:100] test-1_30Ball | Step: 96000. Time Elapsed: 115.010 s Mean Reward: 42.562. Std of Reward: 32.109. Training.
```

링크 모음

[보상함수, 하이퍼 파라미터 설정](#)

[유튜브 유니티 ml-agents 검색](#)

[추천 튜토리얼 블로그](#)

colab으로 학습(둘 다 보면 좋습니다)

[실행 가능한 빌드 파일과 함께 올라간 github](#)

[실제 colab에서 작성한 파일](#)

[공식 Getting-Started.md](#)

[한글로 된 예제들](#)