



The logo features the AWS logo on the left, consisting of the letters "aws" in white with a curved arrow underneath, followed by the words "SUMMIT" and "ONLINE" stacked vertically in white capital letters.

AWS로 강화 학습 쉽게 시작하기

김대근
솔루션즈 아키텍트
AWS Korea

Agenda

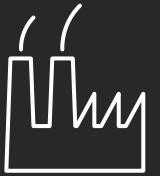
- 강화 학습 기본 개념
- Amazon SageMaker RL
- AWS의 시뮬레이터로 훈련하기

강화 학습(Reinforcement Learning) 기본 개념

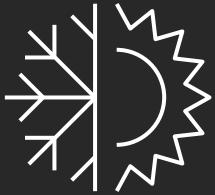
강화 학습은 많은 도메인들에 적용 가능합니다.



로보틱스



산업 시스템 제어



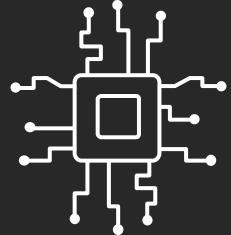
HVAC



자율주행



광고



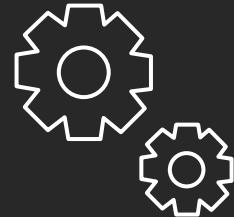
대화 시스템



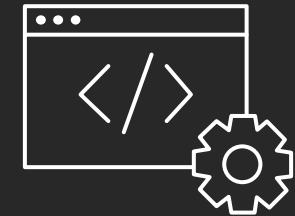
공정



금융



리소스 분배



온라인 컨텐츠 전달

강화 학습(Reinforcement Learning)이란?



- 에이전트 Agent : 학습의 주체
- 환경 Environment : 에이전트를 제외한 나머지
- 상태 State : 환경에 대한 기술
- 행동 Action : 에이전트가 어떤 상태에서 취할 수 있는 행동
- 보상 Reward : 행동의 좋고 나쁨을 즉각적으로 받는 것

강화 학습은 보상 가설 reward hypothesis을 기반으로 합니다.

목표: 누적 보상 기댓값 expected cumulative reward의 극대화

주요 개념: 정책, 에피소드, 리턴

- 정책 Policy: 에이전트가 환경에서 상태를 관찰 후, 수행할 행동을 결정

$$\pi(a|s) = \mathbb{P}[A_t = a | S_t = s]$$

- 에피소드 Episode (i.e, trajectory, rollout): 초기 상태부터 종료 상태까지 에이전트가 거친 상태와 행동의 sequence

$$\tau = (s_0, a_0, s_1, a_1, \dots)$$

- 리턴 Return: 현재 시점 이후의 누적 보상 (Cumulative Reward)

$$G_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \dots = \sum_{i=t}^T \gamma^{i-t} r_i = r_t + \gamma G_{t+1}$$

주요 개념: 가치(Value) 함수와 Q 함수

- 상태 가치 State-Value 함수: 에이전트가 상태 s 로부터 정책 π 에 따라 행동할 때 얻는 누적 보상의 기댓값

$$V_\pi(s) = \mathbb{E}_\pi[G_t | S_t = s]$$

- 행동 가치 Action-Value 함수(Q 함수): 에이전트가 상태 s 에서 행동 a 를 선택하고, 그 이후에 policy π 에 따라 행동할 때 얻는 누적 보상의 기댓값

$$Q_\pi(s, a) = \mathbb{E}_\pi[G_t | S_t = s, A_t = a]$$

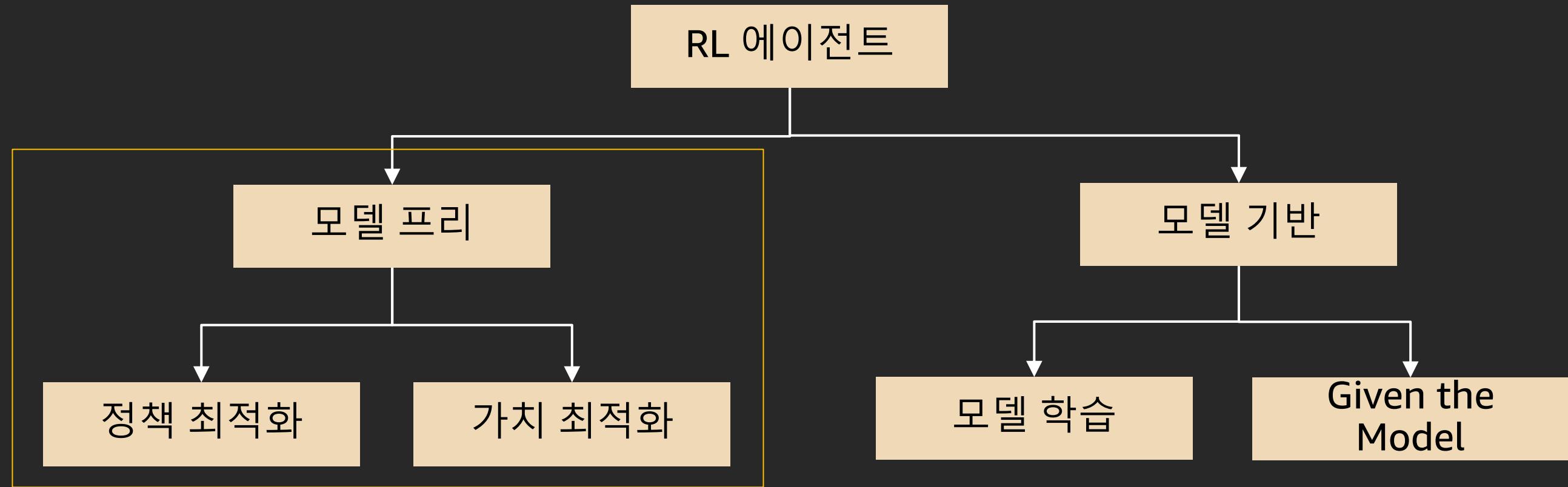
주요 개념: 모델

- 모델 : 환경이 어떻게 될지 예측
 - 가치 함수 예측
 - 다음 상태가 무엇이 될지 예측 (상태 전이 확률 state transition probability)
- (Q) 에이전트가 환경의 상태를 알기가 쉽나요? 아니요
- (Q) 모델 프리로 학습이 가능한가요? 예

강화 학습 vs. 지도 학습

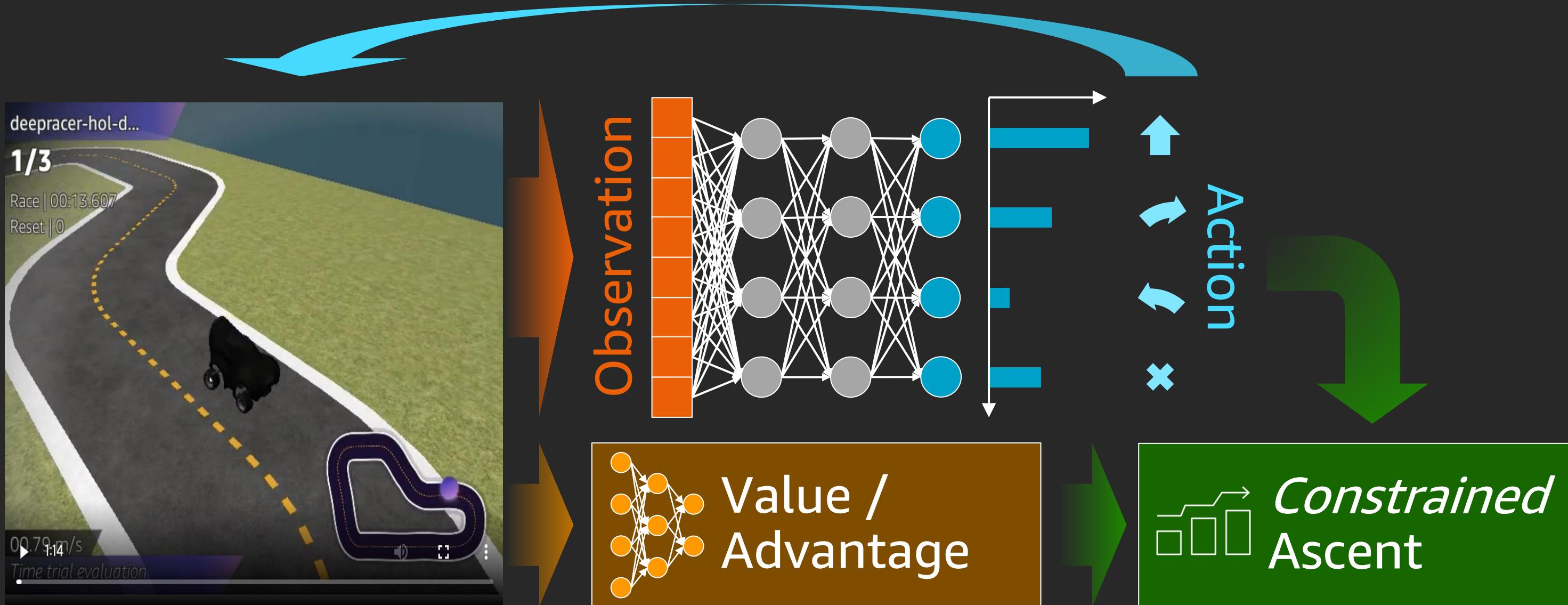
	강화 학습	지도 학습
훈련 데이터	(상태, 행동, 전이 확률, 감가율, 보상)	(X, y)
훈련 파라미터	환경이 어떻게 될지 예측하는 파라미터 (예: 정책)	예측값을 산출하는 파라미터; $\hat{y} = f(X)$
목표	누적 보상 기댓값의 최대화	정답값과 예측값 차이($(y - \hat{y})^2$)의 최소화

모델 프리 vs. 모델 기반



- 장점: 구현이 간단합니다.
- 단점: 샘플이 많이 필요합니다.
- 장점: 복잡한 작업 task 을 잘 해결합니다.
- 단점: 구현이 복잡합니다.

RL 알고리즘: PPO(Proximal Policy Optimization)



Amazon SageMaker RL

Amazon SageMaker RL로 강화 학습을 쉽게 사용해 보세요.

진입 장벽이 있습니다.

RL 에이전트 알고리즘은 구현하기 복잡합니다.

학습 환경의 통합이 어렵습니다.

학습에 계산 비용이 많이 들고 시간이 많이 걸립니다.

시행 착오 및 하이퍼파라미터의 빈번한 튜닝이 필요합니다.

RL을 위한 사전 구축 환경; 수많은 예시

RL 에이전트 알고리즘 지원

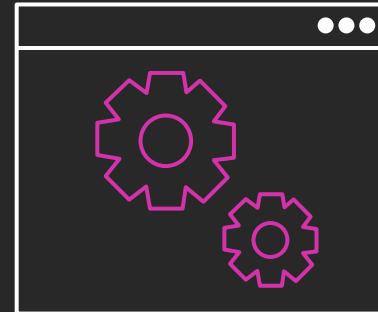
다양한 시뮬레이션 환경을 쉽게 통합

단일/분산 학습; 로컬/원격 환경

디버깅을 위한 로컬 모드; 자동 모델 튜닝

Amazon SageMaker RL

모든 개발자 및 데이터 과학자를 위한 강화 학습



완전 관리



다양한 프레임워크 지원



시뮬레이션 환경에 대한
광범위한 지원

KEY FEATURES

2D & 3D 물리 환경 및
OpenAI Gym 지원

Amazon Sumerian, AWS
RoboMaker 및 오픈 소스 ROOS
(Robotics Operating System)
프로젝트 지원

예제 주피터 노트북 및
튜토리얼 제공

Amazon SageMaker RL과 시뮬레이션

클래식 RL 및 실제 RL 애플리케이션을 위한 End-to-End 예제

로보틱스

산업 제어

HVAC

자율주행

공정

금융

게임

NLP

실제 문제를 모델링하는 RL 환경

AWS 시뮬레이션 프레임워크

AWS Sumerian

Amazon RoboMaker

오픈소스 환경

EnergyPlus

RoboSchool

PyBullet

...

사용자 정의 환경

Bring Your Own

상업 시뮬레이터

MATLAB & Simulink

Open AI Gym

RL 에이전트 알고리즘 구현을 제공하는 RL 툴킷

Intel RL-Coach

DQN

PPO

A3C

Rainbow

...

RL-Ray RLLib

DQN

PPO

IMPALA

A3C

...

Open AI Baselines

DQN

PPO

...

Amazon SageMaker 딥러닝 프레임워크

Tensorflow

MxNet

PyTorch

Chainer



SageMaker supported



Customer BYO

Intel RL Coach 라이브러리

```
[from rl_coach.agents.clipped_ppo_agent import ClippedPPOAgentParameters  
from rl_coach.base_parameters import VisualizationParameters, PresetValidationParameters  
from rl_coach.core_types import TrainingSteps, EnvironmentEpisodes, EnvironmentSteps  
[from rl_coach.environments.gym_environment import GymVectorEnvironment  
from rl_coach.exploration_policies.categorical import CategoricalParameters  
from rl_coach.filters.filter import InputFilter  
from rl_coach.filters.observation.observation_rgb_to_y_filter import ObservationRGBToYFilter  
from rl_coach.filters.observation.observation_stacking_filter import ObservationStackingFilter  
from rl_coach.filters.observation.observation_to_uint8_filter import ObservationToInt8Filter  
from rl_coach.graph_managers.basic_rl_graph_manager import BasicRLGraphManager  
from rl_coach.graph_managers.graph_manager import ScheduleParameters  
from rl_coach.schedules import LinearSchedule  
from rl_coach.base_parameters import DistributedCoachSynchronizationType]
```

빌트인 RL 알고리즘

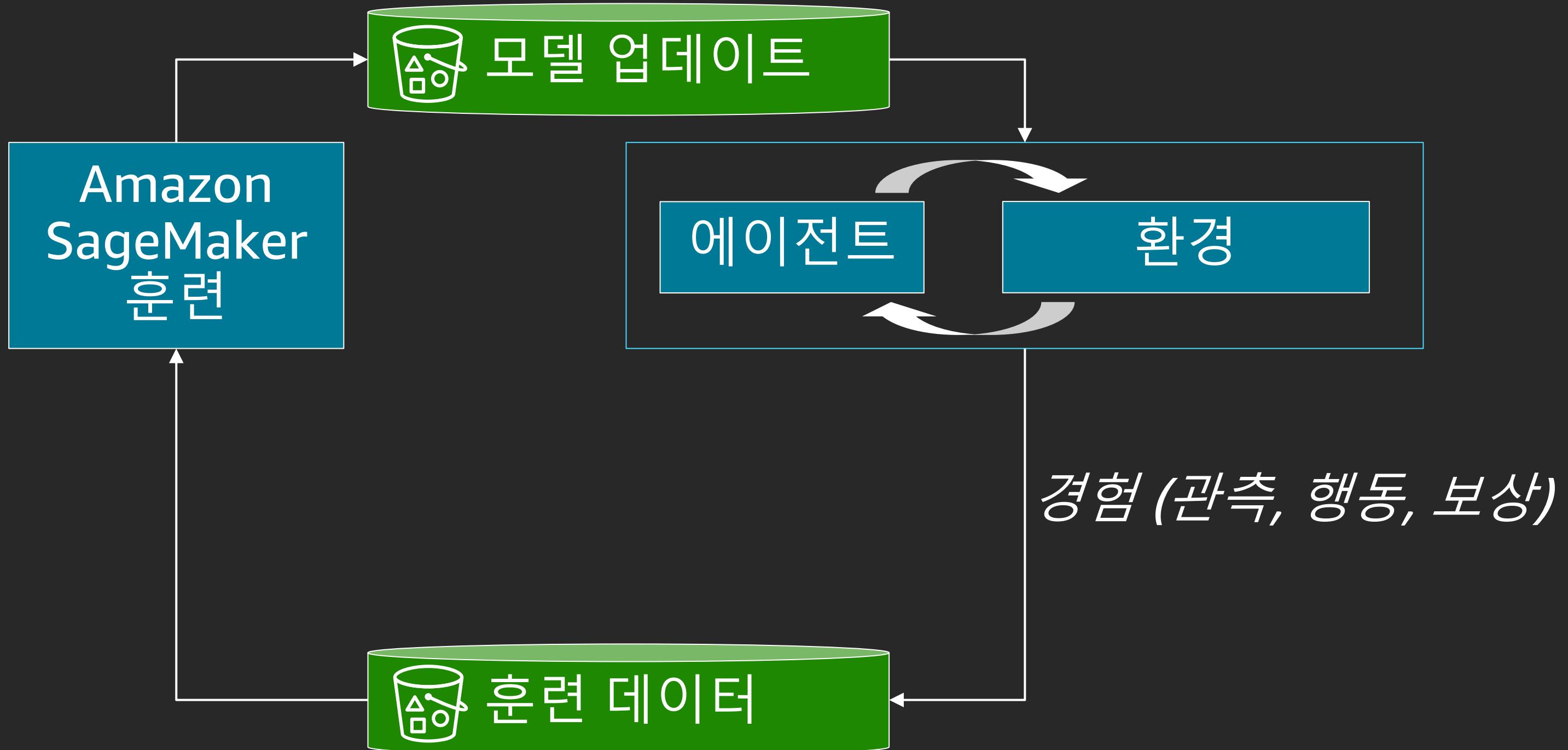
OpenAI Gym 통합

```
import tensorflow as tf from rl_coach.architectures  
import layers from rl_coach.architectures.tensorflow_components.layers  
import Conv2d, Dense from rl_coach.architectures.tensorflow_components import utils
```

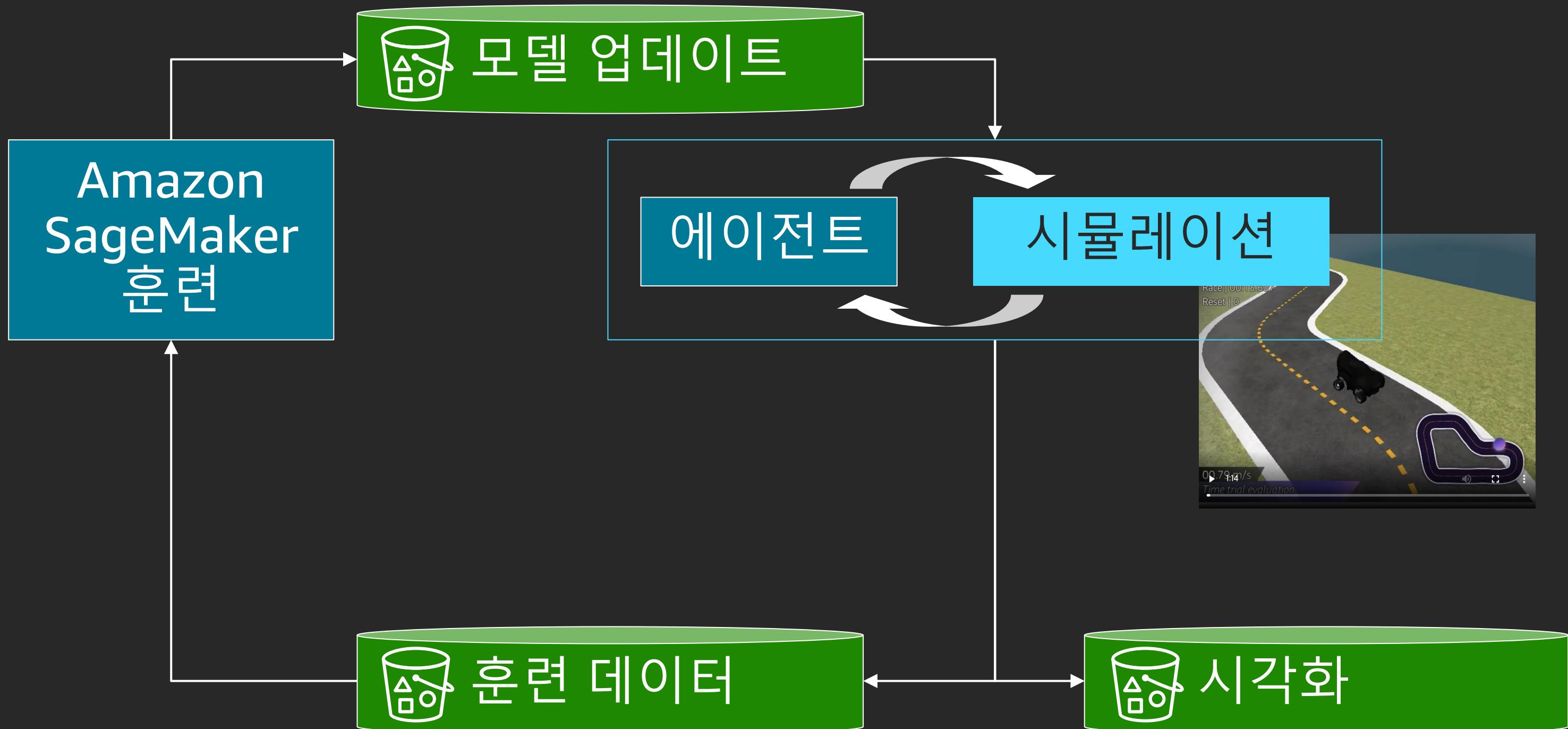
```
conv = tf.layers.conv2d(input_layer, filters=self.num_filters,  
                      kernel_size=self.kernel_size,  
                      strides=self.strides,
```

Intel-optimized
TensorFlow

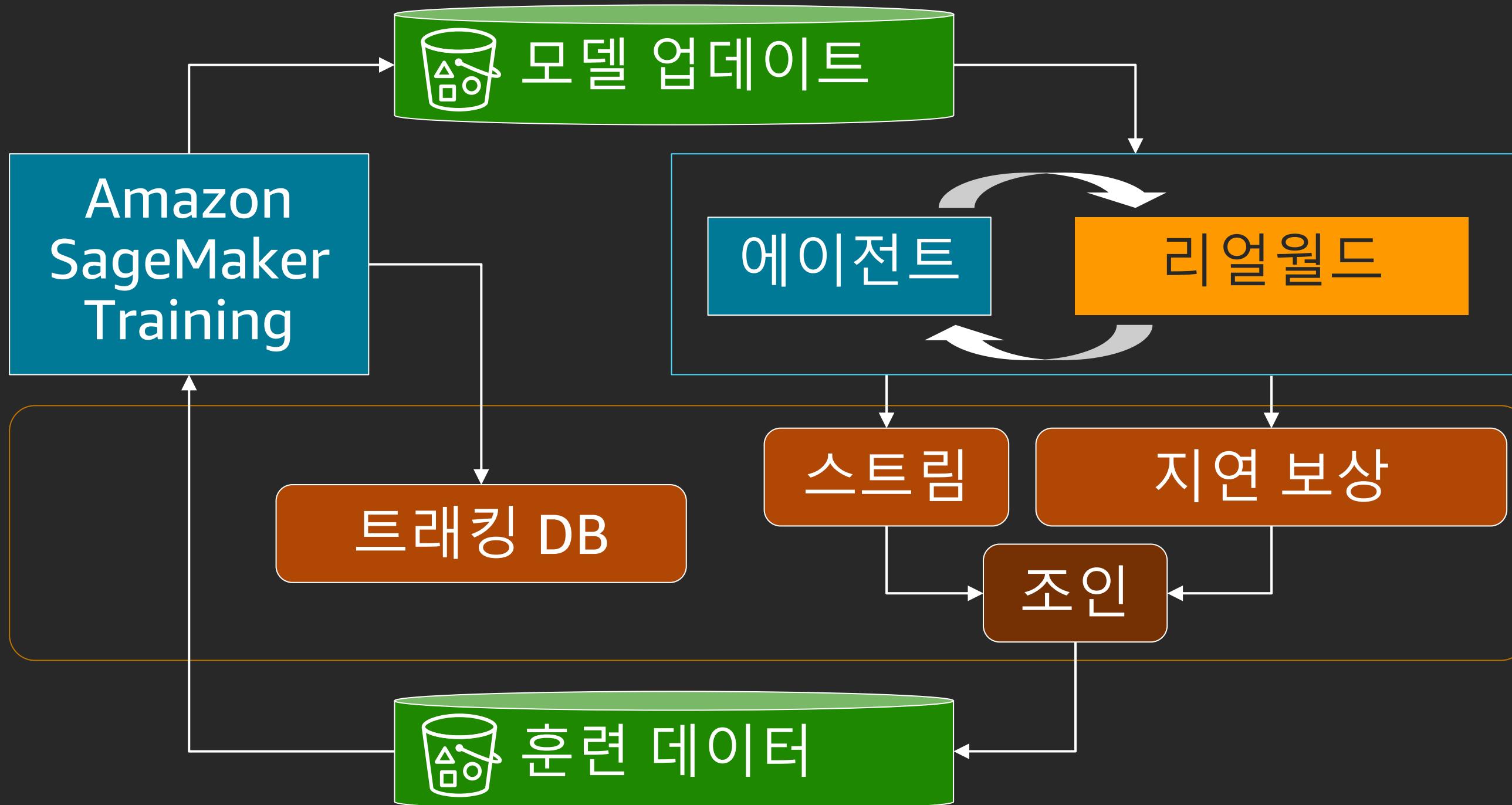
Amazon SageMaker RL



시뮬레이션 기반 RL

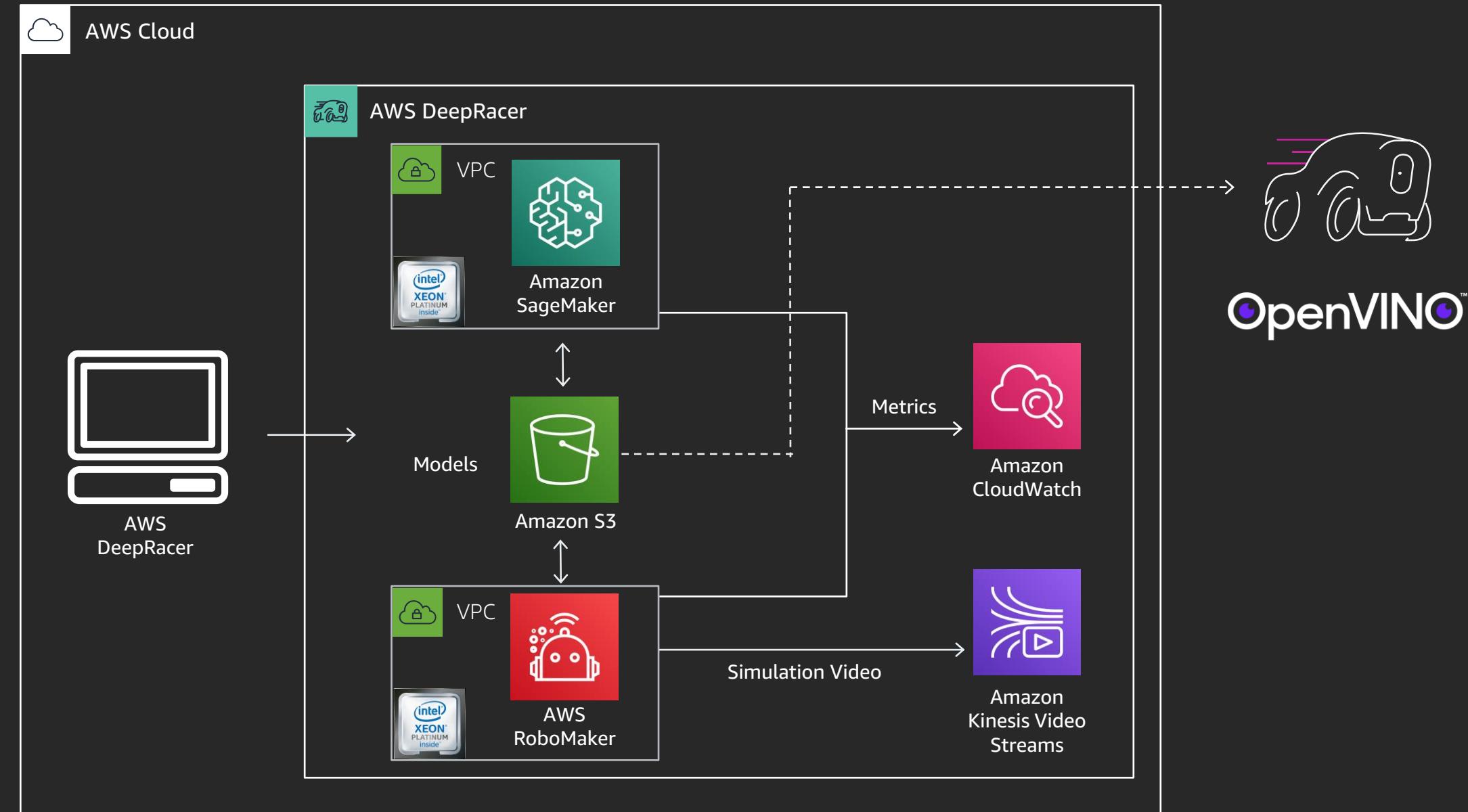


실제 세계 기반 RL

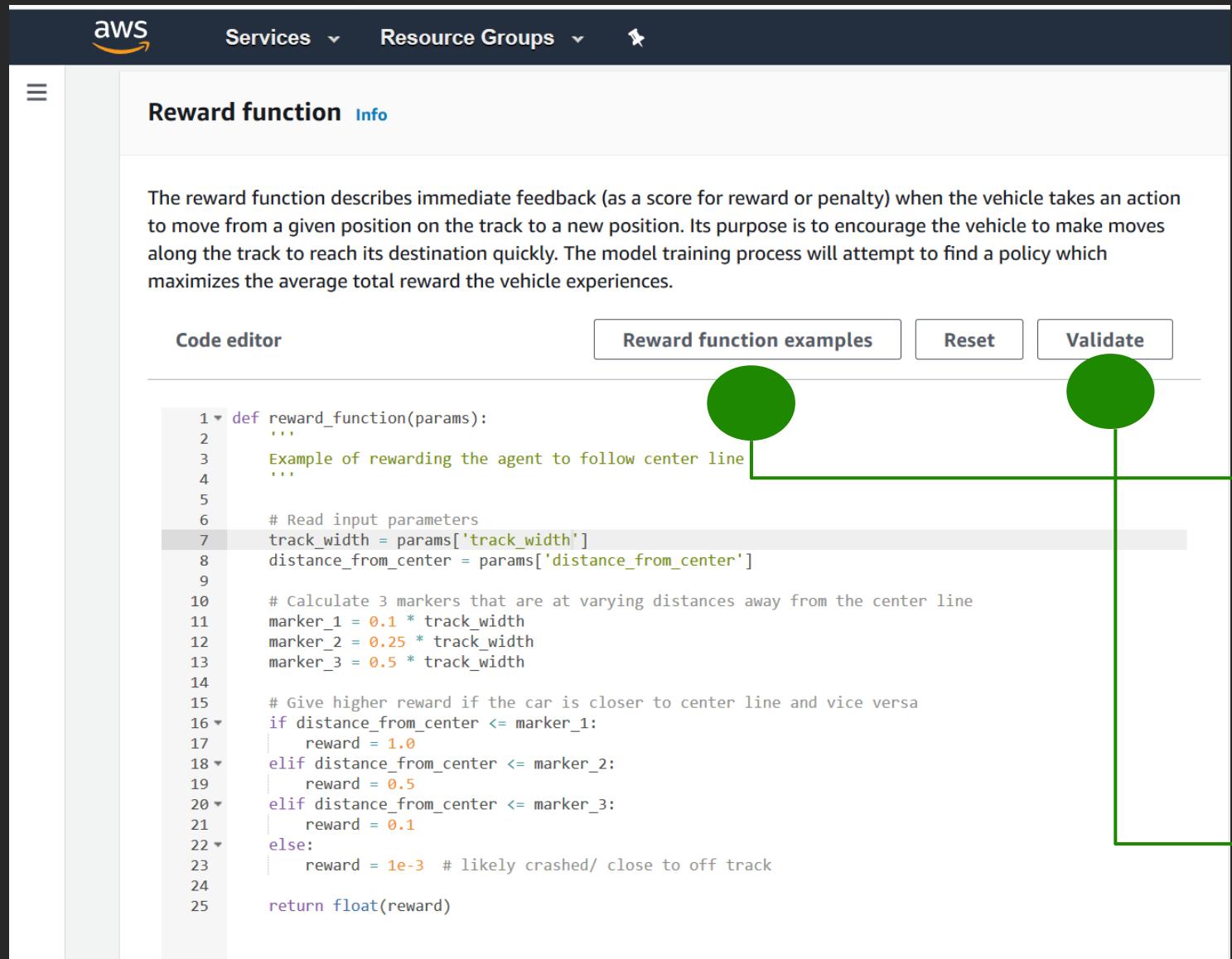


AWS의 시뮬레이터로 학습하기

AWS DeepRacer 시뮬레이터 아키텍처



여러분만의 보상 함수를 쉽게 만들어 보세요.



The reward function describes immediate feedback (as a score for reward or penalty) when the vehicle takes an action to move from a given position on the track to a new position. Its purpose is to encourage the vehicle to make moves along the track to reach its destination quickly. The model training process will attempt to find a policy which maximizes the average total reward the vehicle experiences.

Code editor

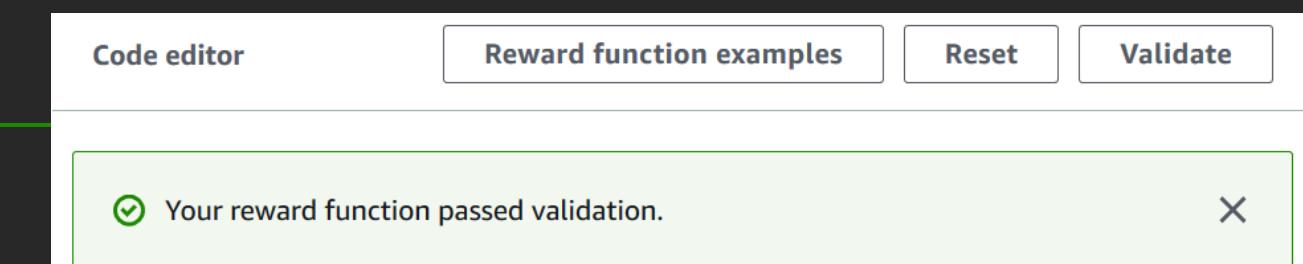
```
1 def reward_function(params):
2     """
3         Example of rewarding the agent to follow center line
4     """
5
6     # Read input parameters
7     track_width = params['track_width']
8     distance_from_center = params['distance_from_center']
9
10    # Calculate 3 markers that are at varying distances away from the center line
11    marker_1 = 0.1 * track_width
12    marker_2 = 0.25 * track_width
13    marker_3 = 0.5 * track_width
14
15    # Give higher reward if the car is closer to center line and vice versa
16    if distance_from_center <= marker_1:
17        reward = 1.0
18    elif distance_from_center <= marker_2:
19        reward = 0.5
20    elif distance_from_center <= marker_3:
21        reward = 0.1
22    else:
23        reward = 1e-3 # likely crashed/ close to off track
24
25    return float(reward)
```

Reward function examples Reset Validate

코드 편집: Python 3 문법

4가지 보상 함수 예제

AWS Lambda를 통한 코드 검증



console.aws.amazon.com/deepracer/home?region=us-east-1#getStarted

Amazon Web Serv... Isengard Daekeun Wiki Amazon WorkDocs Outlook Web App Phone Tool Study Guide Trouble Ticketing Concur SSO GitHub - awskrug/... QnA Training Korea HR

daekeun@amazon.com - 143656149352 / Admin (Not Production Account)

AWS DeepRacer

Racing League AWS Virtual Circuit Community races Reinforcement learning Get started Your models Your garage Resources About the league Schedules & standings Rules & prizes Developer guide Tips & tricks Forum Slack channel

AWS DeepRacer > Get started

Get started with reinforcement learning

Overview

Learn RL: Set up account resources. Learn the basics of reinforcement learning.

Create model: Define the reward function, hyperparameters and action space.

Train model: Observe the agent interact with the environment in simulation.

Evaluate model: Watch the agent drive itself in a simulated track environments.

Join DeepRacer League: See how your model stacks up. No cost to enter, unlimited model submissions.

Step 0: Create account resources (Required / ~5 mins) Info

To get you set up, AWS DeepRacer needs to create account resources so that you can train and evaluate the models you build. Setting up account resources will not incur any charges on your account.

- ✓ You have valid IAM roles
- ✓ You have a valid AWS DeepRacer resources stack

Having issues? Try resetting the resources created by AWS DeepRacer to start again from a clean slate. Info

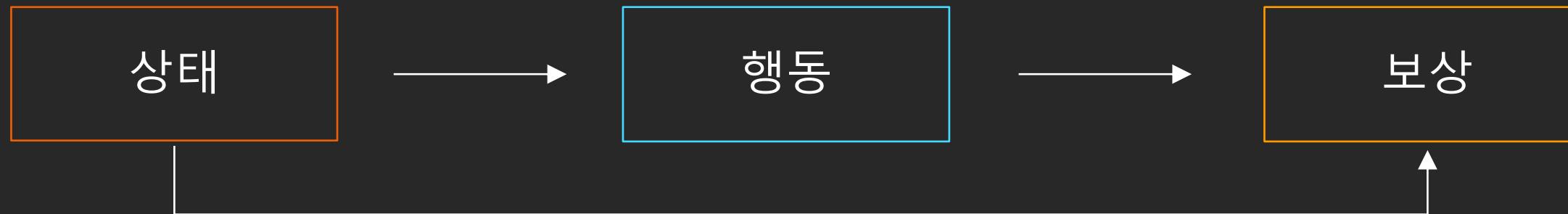
[Reset resources](#)

https://console.aws.amazon.com/deepracer/home?region=us-east-1#garage

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Contextual MAB vs. Full RL

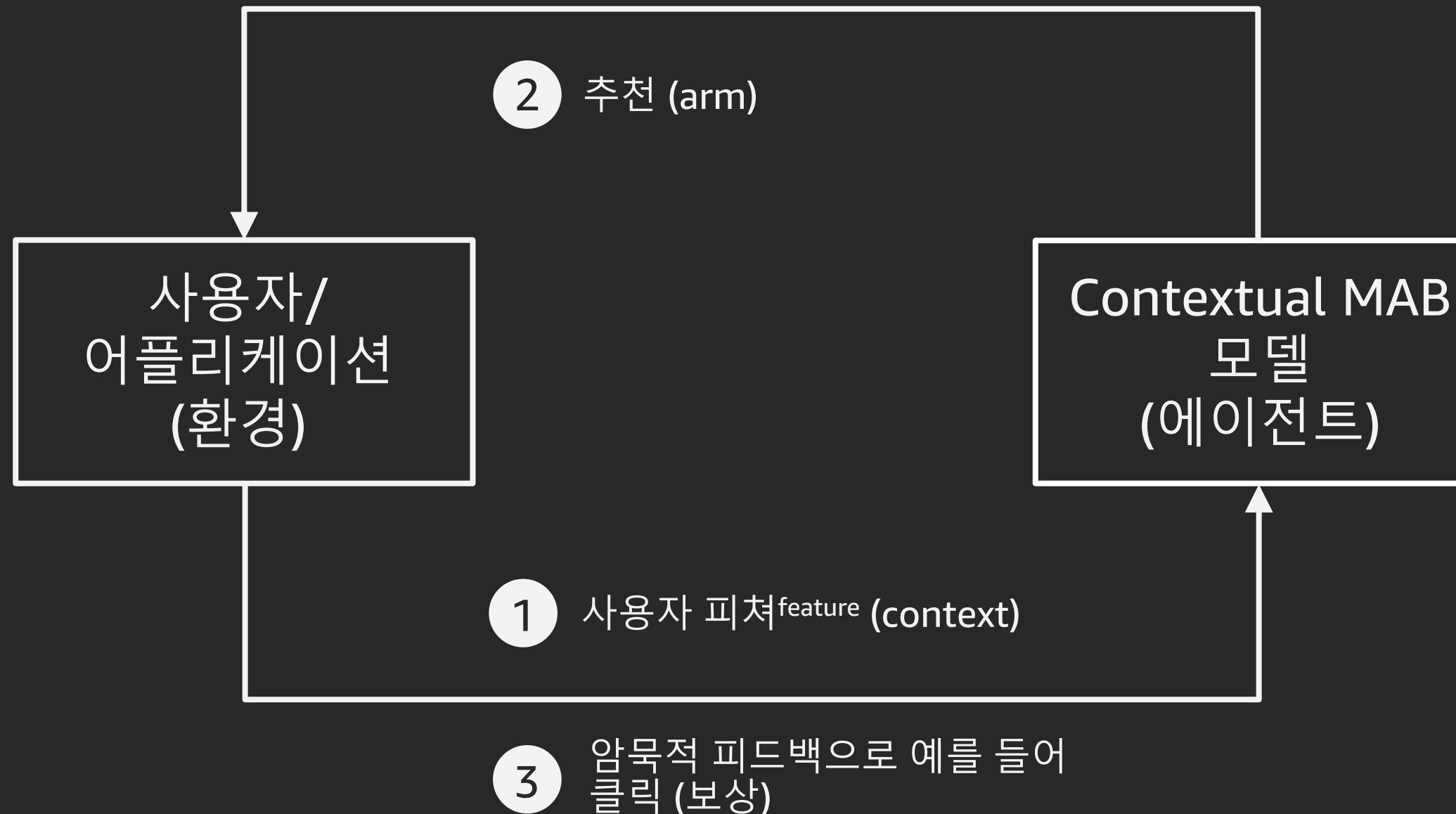
- Contextual Multi-Armed Bandit: 랜덤하게 상태(i.e. 밴딧^{bandit})가 주어지며, 상태는 변하지 않습니다.



- Full RL (Policy gradient, PPO, etc.)



Contextual MAB로 추천 엔진 구축하기



공식 샘플 예제

https://github.com/awslabs/amazon-sagemaker-examples/tree/master/reinforcement_learning

Bandits for live A/B testing

Train with offline data

*Autonomous
Robots*

HVAC Energy Optimization

Neural Network Compression

Autoscaling

Branch: master		amazon-sagemaker-examples / reinforcement_learning
yijiezz Add batch RL example notebook (#926)		
..		
bandits_statlog_vw_customEnv		Add Contextual Bandits example (#857)
common		Add batch RL example notebook (#926)
rl_cartpole_batch_coach		Add batch RL example notebook (#926)
rl_cartpole_coach		Reuse common code/config (#750)
rl_deepracer_robomaker_coach_ga...		Sagemaker public notebook consistant with
rl_hvac_coach_energyplus		Fix HVAC EnergyPlus RL Hosting (#890)
rl_knapsack_coach_custom		Reuse common code/config (#750)
rl_managed_spot_cartpole_coach		Adding a managed spot training example (#
rl_mountain_car_coach_gymEnv		Reuse common code/config (#750)
rl_network_compression_ray_custom		Reuse common code/config (#750)
rl_objecttracker_robomaker_coach...		Reuse common code/config (#750)
rl_portfolio_management_coach_c...		Reuse common code/config (#750)
rl_predictive_autoscaling_coach_cu...		Reuse common code/config (#750)
rl_roboschool_ray		Support restoring checkpoint in ray; Add pa
rl_roboschool_stable_baselines		Reuse common code/config (#750)
rl_tic_tac_toe_coach_customEnv		Add SageMaker RL Tic-Tac-Toe Example (#
rl_traveling_salesman_vehicle_routi...		fixes #775 (#777)

샘플 예제 (한국어화)

- <http://bit.ly/sagemaker-rl-kr>
 - *deepracer_rl.ipynb*: SageMaker와 RoboMaker로 분산 DeepRacer RL 훈련
 - *rl_roboschool_stable_baselines.ipynb*: Stable baseline으로 Roboschool 시뮬레이션 훈련
 - *bandits_movielens_testbed.ipynb*: Contextual Bandits 기반 movielens 영화 추천
 - *rl_objecttracker_coach_robomaker.ipynb*: 분산 Object Tracker RL 훈련
 - *rl_predictive_autoscaling_coach_customEnv.ipynb*: SageMaker RL을 활용한 오토스케일링
 - *rl_roboschool_ray.ipynb*: 물리 로봇의 Roboschool 시뮬레이션
 - *rl_roboschool_ray_distributed.ipynb*: 다중 노드들 간 분산 RL로 Roboschool 시뮬레이션
 - *rl_roboschool_ray_automatic_model_tuning.ipynb*: RL 기반 자동 하이퍼파라메터 튜닝

여러분의 소중한 피드백을 기다립니다!

강연 평가 및 설문 조사에 참여해 주세요.

감사합니다