Adaptive Learning in Motion:

Harnessing Cloud-Based AI for Simulating Smart Navigation

School of Technology & Computing

Masters of Artificial Intelligence

Verónica Elze

Vidhyalakshmi Amarnath

Masters of Computer Science

Jayasudha Kalamegam

Honorine Ndom Ndzah

CityU of Seattle

Agenda

- Goal, Motivation & Use
- Approach
- Background
- AI Process
- Demonstration
- Migration
- Cloud Service(s)
- Architecture
- Model Selection
- Metrics & Visualizations
- Observations & Conclusion
- Future Potential Work

CityU of Seattle

Goal, Motivation, & Use

Navigate Dynamic Environments

Low-cost autonomous systems

Adaptation/Anticipation over Reaction

CityU of Seattle

Approach

Research Models & Services

Review Relevant Literature

Analyze Similar Projects

Manual Build & Test Prototype(s)

Amazon Web Service & Metrics

Background

Simulation-Based Navigation Testing

Reinforcement Learning for Simulated Environments

Cloud Computing for AI Scalability

CityU of Seattle

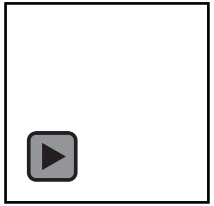
AI Local Process

BUILD
- ✓ Define model (PPO)
- ✓ Create custom simulation environment
- ✓ Preload essential configurations

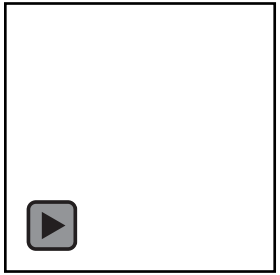
TRAIN
- ✓ Execute iterations
- ✓ Monitor metrics
- ✓ Adjust parameters

TEST
- ✓ Run in virtual environment
- ✓ Measure navigation accuracy
- ✓ Analyze performance in various scenarios

Small Grid
Get to Goal



Medium Grid
Avoid Dynamic Objects
Get to Goal

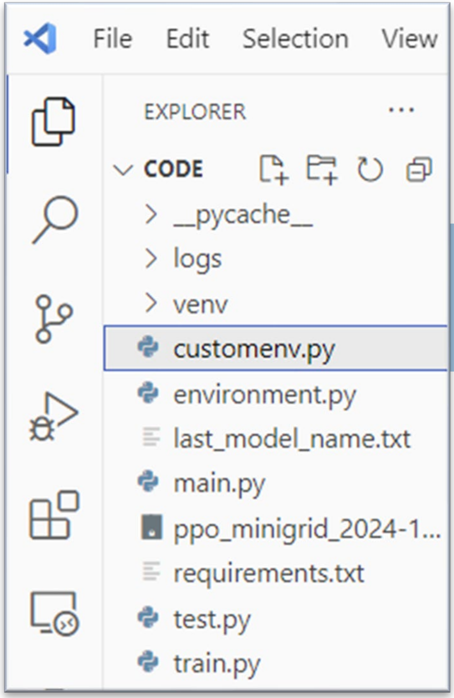


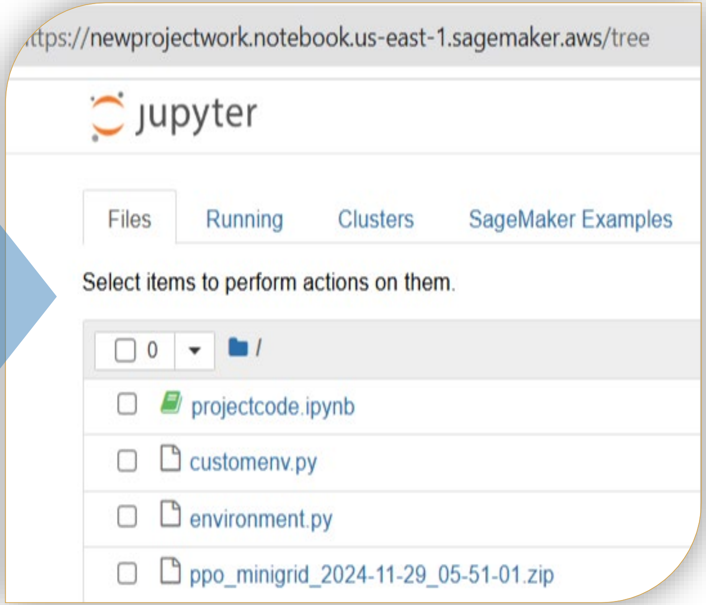
CityU of Seattle

Demonstration

CityU of Seattle

Migration Local to Cloud





CityU of Seattle

Amazon Web Services

Compute Power
SageMaker
Train RL models

Scalable Storage
S3
Handle data

Security Measures
IAM
Ensure data privacy

Logs
CloudWatch
Monitor resources

SimSpace Weaver
Simulate in 3D

IoT
Manage Devices

CityU of Seattle

Project Architecture Flow

OpenAI Gym
- Generate Data
- Simulate Environment

AWS IAM
- Control Access
- Protect Resources

Amazon S3
- Store:
  - Models
  - Actions
  - States
  - Rewards

Amazon SageMaker
- Train RL Model
- Refine Agent

Amazon CloudWatch
- Monitor:
  - Progress
  - Failures
  - Resources

CityU of Seattle

Model Selection

Choosing Optimal RL Models for Dynamic Environments

Proximal Policy Optimization (PPO)

- ✓ High sample efficiency
- ✓ Proven effectiveness in dynamic scenarios

✓ Exploration
✓ Exploitation

✓ MiniGrid
✓ Actions

✓ Powerful
✓ Simple

Metrics

Training

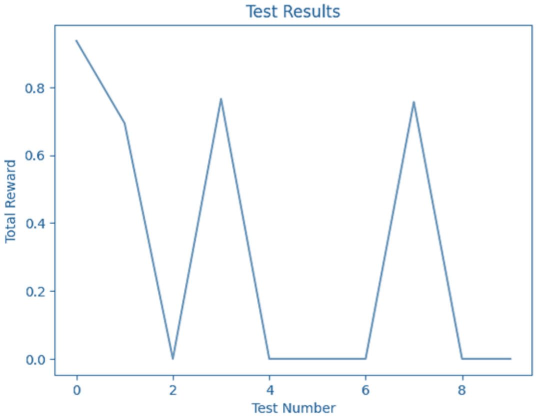
Cumulative Rewards: performance improvement

Convergence Time: RL model stabilization time

Simulation

Success Rate: successful navigations %

Obstacle Avoidance: Accuracy avoiding obstacles




```
Episode finished after 7 steps with total reward: 0.9369999999999999
Episode finished after 34 steps with total reward: 0.694
Episode finished after 100 steps with total reward: 0
Episode finished after 26 steps with total reward: 0.766
Episode finished after 100 steps with total reward: 0
Episode finished after 100 steps with total reward: 0
Episode finished after 100 steps with total reward: 0
Episode finished after 27 steps with total reward: 0.757
Episode finished after 100 steps with total reward: 0
Episode finished after 100 steps with total reward: 0
```

Efficiency

Training Time per Epoch: training speed

Simulation Runtime: computational efficiency

Robustness

Generalization: performance in new environments

Failure Rate: undesirable outcomes frequency

CityU of Seattle

Visualizations

Training Iteration Progress

rollout/ep_len_mean Average episode length during rollouts

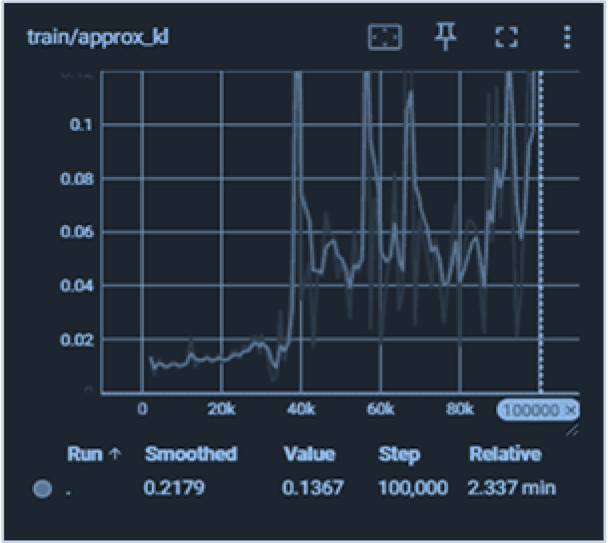
rollout/ep_rew_mean Average reward per episode

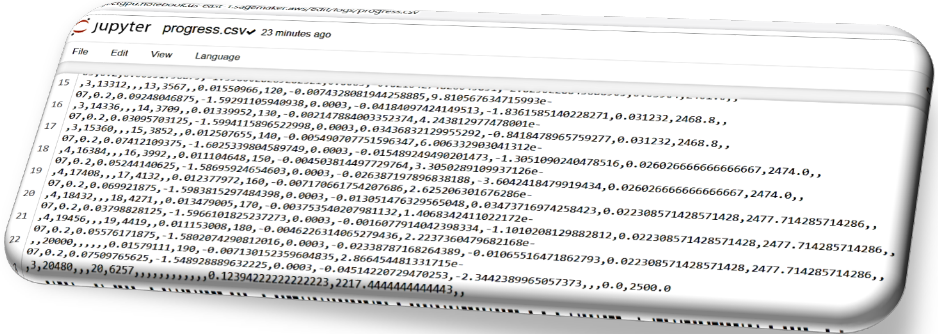
train/approx_kl Kullback-Leibler divergence for policy updates

train/explained_variance Measures how well the value function predicts rewards

train/policy_gradient_loss & **value_loss** Policy & value functions performance indicators

time/fps & **time/total_timesteps** Performance & efficiency indicators during training




jupyter progress.csv 23 minutes ago

File Edit View Language

```
,3,13312,,,13,3567,,0.01550966,120,-0.0074328081944258885,9.810567634715993e-
07,0.2,0.09248046875,-1.59291105940938,0.0003,-0.04184097424149513,-1.8361585140228271,0.031232,2468.8,,
,3,14336,,,14,3709,,0.01339952,130,-0.002147884003352374,4.243812977478001e-
07,0.2,0.03095703125,-1.5994115896522998,0.0003,0.03436832129955292,-0.8418478965759277,0.031232,2468.8,,
,3,15360,,,15,3852,,0.012507655,140,-0.005490707751596347,6.006332903041312e-
07,0.2,0.07412109375,-1.6025339804589749,0.0003,-0.015489249490201473,-1.3051090240478516,0.026026666666666667,2474.0,,
,4,16384,,,16,3992,,0.011104648,150,-0.004503814497729764,3.3050289109937126e-
07,0.2,0.05244140625,-1.58695924654603,0.0003,-0.026387197896838188,-3.6042418479919434,0.026026666666666667,2474.0,,
,4,17408,,,17,4132,,0.012377972,160,-0.007170661754207686,2.625206301676286e-
07,0.2,0.069921875,-1.5983815297484398,0.0003,-0.013051476329565048,0.03473716974258423,0.022308571428571428,2477.714285714286,,
,4,18432,,,18,4271,,0.013479005,170,-0.003753540207981132,1.4068342411022172e-
07,0.2,0.03798828125,-1.5966101825237273,0.0003,-0.0016077914042398334,-1.1010208129882812,0.022308571428571428,2477.714285714286,,
,4,19456,,,19,4419,,0.011153008,180,-0.0046226314065279436,2.2237360479682168e-
07,0.2,0.05576171875,-1.5802074290812016,0.0003,-0.02338787168264389,-0.01065516471862793,0.022308571428571428,2477.714285714286,,
,,20000,,,,,,0.01579111,190,-0.007130152359604835,2.866454481331715e-
07,0.2,0.07509765625,-1.548928889632225,0.0003,-0.04514220729470253,-2.3442389965057373,,,0.0,2500.0
,3,20480,,,20,6257,,,,,,,,,,,,,0.12394222222222223,2217.4444444444443,,
```

CityU of Seattle

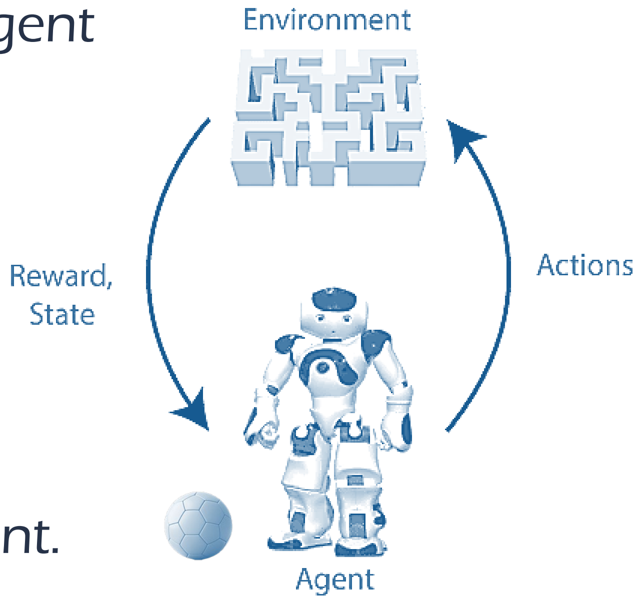
Observations

SPEED	Early slow, Quick later
ACCURACY	Count Steps to Ball & Goal
BEHAVIOR	Custom Kick Action & Environment
EFFICIENCY	Improvement over time

CityU
of Seattle

Conclusions

- Trained a Proximal Policy Optimization (PPO) agent to successfully navigate complex environments with obstacles.
- Demonstrated the ability to learn and adapt behavior based on environmental feedback.
- Achieved efficient performance with limited computational resources, utilizing AWS cloud services for scalable training & data management.



CityU of Seattle

Future Potential Work

3D Simulation Enhancements
Upgrade to 3D simulations with tools like **AWS SimSpace Weaver** for realistic testing.

Dynamic Obstacle Interaction
Train RL models to handle moving players and unpredictable dynamics.

ML Model Refinements
Use advanced RL algorithms (e.g., DDPG, Multi-Agent RL) for better adaptability.

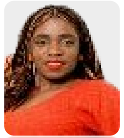
CityU of Seattle

Thank you!

Verónica Elze
MS of AI

Vidhyalakshmi Amarnath
MS in AI

Jayasudha Kalamegam
MS in CS



Honorine Ndom Ndzah
MS in CS

OpenAI. (2024). *Reinforcement learning project graphic: Smart soccer ball navigating a maze*. Generated using DALL·E.

CityU
of Seattle