February 20, 2020

Reinforcement Learning Penguins (Part 4/4) | Unity ML-Agents

TRAINING AND INFERENCE

Before you can train your penguins, you will need to set up a couple configuration files. The first one (the trainer_config.yaml file) is always necessary, but the second (the penguin.yaml file) is only needed if you want to use curriculum to assist in training. We are using curriculum to gradually increase the difficulty for our penguins and speed up training significantly. In this case, we initially reward the penguin for catching a fish and coming within a large radius of the baby. Then the curriculum settings will gradually shrink the acceptable feeding radius as the penguin gets better at the task. Without the curriculum, the penguins would take a lot longer to learn.

Note: In the steps below, make sure you spell “PenguinLearning” correctly and that it matches the Behavior Name added to the Agent when you built the Scene. For example, if your Behavior Name is “PenguinLearning” and your curriculum file is “PenguinLearn.json,” training will fail.

TRAINER CONFIG YAML

In this section, you’ll add configuration info that the ML-Agents training program will use. Some of these are hyperparameters — a term that may be familiar to you if you’ve done some deep learning in the past — and others

are settings specific to ML-Agents. While optional for this

Inside the ml-agents directory you downloaded from GitHub, there is a *config* directory. Inside that, you will find a text file called *trainer_config.yaml.*

- Open *config\trainer_config.yaml*
- Add the following lines to the end of the file:

```
PenguinLearning:
    summary_freq: 5000
    time_horizon: 128
    batch_size: 128
    buffer_size: 2048
    hidden_units: 256
    beta: 1.0e-2
    max_steps: 1.0e6
```

If you look at the top of the file, there are lots of default parameters. Anything not explicitly set in the PenguinLearning section will use the default value. If you would like to know more about these parameters, read the Training-PPO documentation.

CURRICULUM

In this section, you'll set up a curriculum for the penguin agents to learn increasingly difficult tasks.

Inside the *config* directory, you'll find a *curricula* directory.

- Open the *curricula* directory.
- Create a new folder inside, called *penguin*.
- Create a new text file called *penguin.yaml* inside the *config\curricula* directory.
- Open *penguin.yaml*.
- Add the following lines of text:

```
PenguinLearning:
```

```
min_lesson_length: 80
signal_smoothing: true
parameters:
  fish_speed: [0.0, 0.0, 0.0, 0.0, 0.0, 0
  feed_radius: [6.0, 5.0, 4.0, 3.0, 2.0,
```

This tells the training code to update fish_speed and feed_radius as the penguins pass certain reward thresholds.

- The starting feed_radius is 6.0 and fish_speed is 0.0.
- Once the average reward earned by the training penguins is greater than -0.1, the feed_radius will decrease to 5.0.
- Once the average reward reaches 0.7, the feed_radius decreases to 4.0, and so on.
- The "min_lesson_length" parameter specifies a minimum number of lessons that must happen before changing, so even if the agent were to get lucky with a high score, it would prevent the difficulty from * increasing immediately.

TRAINING WITH PYTHON

In this section, you'll train the penguin agents using a Python program called mlagents-learn. You will need to install Anaconda and then install the ML-Agents python libraries, as detailed in the Anaconda Setup tutorial. This program is part of the ML-Agents project and makes training much easier than writing your own training code from scratch.

It is important that the version of your training code matches the version in your Unity project. If you updated one, but not the other, training will fail.

- Open Anaconda prompt.

- Activate your ml-agents environment.

- Type in the following command:

```
mlagents-learn config/trainer_config.yaml -
```

Here’s a breakdown of the different parts of the command:

`mlagents-learn:` The Python program that runs training

`config/trainer_config.yaml:` A relative path to the configuration file (this can also be a direct path)

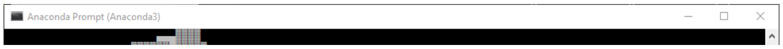
`--curriculum config/curricula/penguin:` A relative path to the curriculum files (this can also be a direct path)

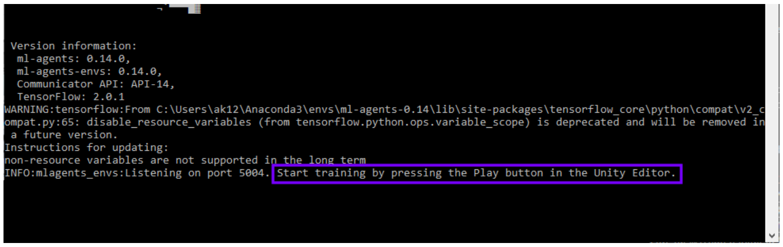
`--run-id penguin_01:` A unique name we choose to give this round of training (you can make this whatever you want)

`--train:` Instructs the program to train the agents rather than just testing them

The reason that a folder is passed in for the curriculum, rather than the file directly, is that it is possible for a project to have multiple curriculum files for different Behavior Names. This project only has one.

- Run the command.
- When prompted, press **Play** in the Unity Editor to start training **(Figure 01)**.





```
 Version information:
  ml-agents: 0.14.0,
  ml-agents-envs: 0.14.0,
  Communicator API: API-14,
  TensorFlow: 2.0.1
WARNING:tensorflow:From C:\Users\ak12\Anaconda3\envs\ml-agents-0.14\lib\site-packages\tensorflow_core\python\compat\v2_c
ompat.py:65: disable_resource_variables (from tensorflow.python.ops.variable_scope) is deprecated and will be removed in
 a future version.
Instructions for updating:
non-resource variables are not supported in the long term
INFO:mlagents_envs:Listening on port 5004. Start training by pressing the Play button in the Unity Editor.
```

Figure 01: Anaconda prompt window: mlagents-learn is waiting for you to press the Play button

The penguins will be moving very fast and the frame rate will be choppy because it’s running at 100x speed, which is likely faster than your frame rate.

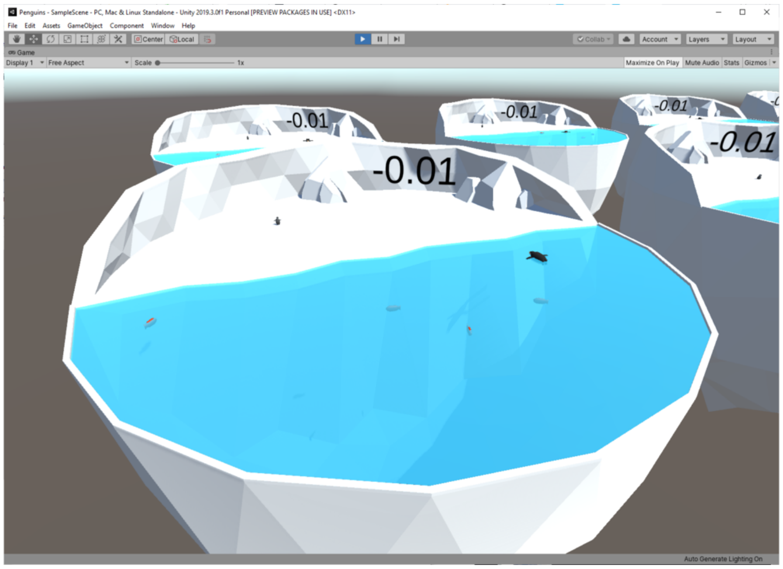
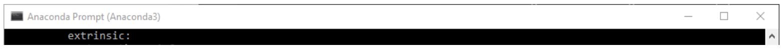

Penguins - SampleScene - PC, Mac & Linux Standalone - Unity 2019.3.0f1 Personal [PREVIEW PACKAGES IN USE] <DX11>
File Edit Assets GameObject Component Window Help
Center Local
Collab Account Layers Layout
Game
Display 1 Free Aspect Scale 1x
Maximize On Play Mute Audio Stats Gizmos
-0.01
-0.01
-0.01
-0.01
-0.01
Auto Generate Lighting On

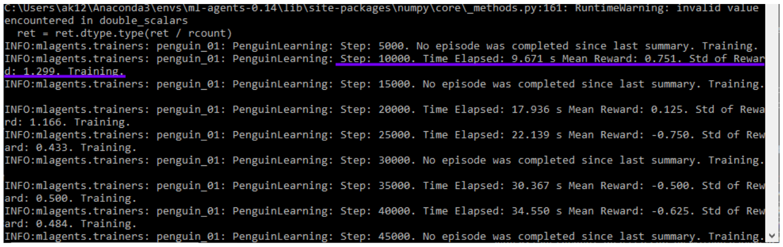
Figure 02: Training started

As the training proceeds, you will get periodic updates **(Figure 03)**. Each of these will include:

- Step: The number of timesteps that have elapsed
- Time Elapsed: How much time the training has been running (in real-world time)
- Mean Reward: The average reward (since the last update)
- Std of Reward: The standard deviation of the reward (since the last update)



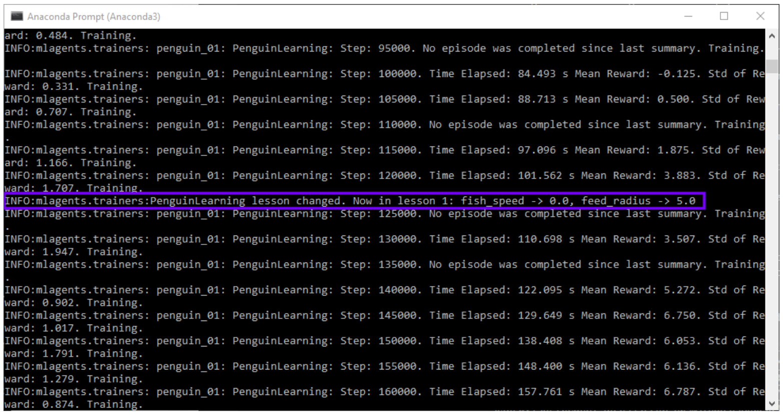
```
        extrinsic:
```



```
C:\Users\ak12\Anaconda3\envs\ml-agents-0.14\lib\site-packages\numpy\core\_methods.py:161: RuntimeWarning: invalid value
encountered in double_scalars
  ret = ret.dtype.type(ret / rcount)
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 5000. No episode was completed since last summary. Training.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 10000. Time Elapsed: 9.671 s Mean Reward: 0.751. Std of Rewar
d: 1.299. Training.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 15000. No episode was completed since last summary. Training.

INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 20000. Time Elapsed: 17.936 s Mean Reward: 0.125. Std of Rewa
rd: 1.166. Training.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 25000. Time Elapsed: 22.139 s Mean Reward: -0.750. Std of Rew
ard: 0.433. Training.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 30000. No episode was completed since last summary. Training.

INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 35000. Time Elapsed: 30.367 s Mean Reward: -0.500. Std of Rew
ard: 0.500. Training.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 40000. Time Elapsed: 34.550 s Mean Reward: -0.625. Std of Rew
ard: 0.484. Training.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 45000. No episode was completed since last summary. Training.
```

Figure 03: Anaconda prompt window: periodic training updates

When the mean reward passes a threshold specified in your curriculum file, you will see an Academy Reset in the training logs **(Figure 04)**. This will update the reset parameters.



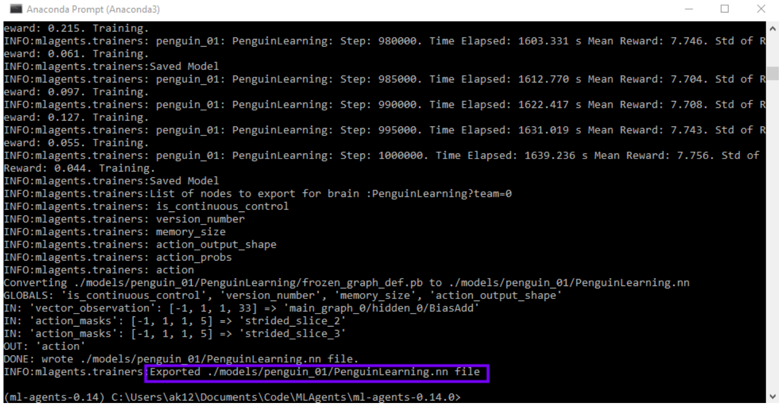
```
ard: 0.484. Training.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 95000. No episode was completed since last summary. Training.

INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 100000. Time Elapsed: 84.493 s Mean Reward: -0.125. Std of Re
ward: 0.331. Training.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 105000. Time Elapsed: 88.713 s Mean Reward: 0.500. Std of Rew
ard: 0.707. Training.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 110000. No episode was completed since last summary. Training
.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 115000. Time Elapsed: 97.096 s Mean Reward: 1.875. Std of Rew
ard: 1.166. Training.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 120000. Time Elapsed: 101.562 s Mean Reward: 3.883. Std of Re
ward: 1.707. Training.
INFO:mlagents.trainers:PenguinLearning lesson changed. Now in lesson 1: fish_speed -> 0.0, feed_radius -> 5.0
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 125000. No episode was completed since last summary. Training
.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 130000. Time Elapsed: 110.698 s Mean Reward: 3.507. Std of Re
ward: 1.947. Training.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 135000. No episode was completed since last summary. Training
.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 140000. Time Elapsed: 122.095 s Mean Reward: 5.272. Std of Re
ward: 0.902. Training.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 145000. Time Elapsed: 129.649 s Mean Reward: 6.750. Std of Re
ward: 1.017. Training.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 150000. Time Elapsed: 138.408 s Mean Reward: 6.053. Std of Re
ward: 1.791. Training.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 155000. Time Elapsed: 148.400 s Mean Reward: 6.136. Std of Re
ward: 1.279. Training.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 160000. Time Elapsed: 157.761 s Mean Reward: 6.787. Std of Re
ward: 0.874. Training.
```

Figure 04: Anaconda prompt window: mean reward passing thresholds

Eventually, your training should reach the maximum curriculum difficulty and the mean reward will no longer increase when the penguins can't pick up fish any faster. At this point, you can stop training early by pressing the **Play** button in the Editor.

The training will export a *PenguinLearning.nn* file that represents a trained neural network for your penguin **(Figure 05)**.

Note: This neural network will only work for your current penguins. If you change the observations in the CollectObservations() function in the PenguinAgent script or RayPerceptionSensorComponent3D, the neural network will in the PenguinAgent script.



```
eward: 0.215. Training.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 980000. Time Elapsed: 1603.331 s Mean Reward: 7.746. Std of R
eward: 0.061. Training.
INFO:mlagents.trainers:Saved Model
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 985000. Time Elapsed: 1612.770 s Mean Reward: 7.704. Std of R
eward: 0.097. Training.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 990000. Time Elapsed: 1622.417 s Mean Reward: 7.708. Std of R
eward: 0.127. Training.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 995000. Time Elapsed: 1631.019 s Mean Reward: 7.743. Std of R
eward: 0.055. Training.
INFO:mlagents.trainers: penguin_01: PenguinLearning: Step: 1000000. Time Elapsed: 1639.236 s Mean Reward: 7.756. Std of
Reward: 0.044. Training.
INFO:mlagents.trainers:Saved Model
INFO:mlagents.trainers:List of nodes to export for brain :PenguinLearning?team=0
INFO:mlagents.trainers: is_continuous_control
INFO:mlagents.trainers: version_number
INFO:mlagents.trainers: memory_size
INFO:mlagents.trainers: action_output_shape
INFO:mlagents.trainers: action_probs
INFO:mlagents.trainers: action
Converting ./models/penguin_01/PenguinLearning/frozen_graph_def.pb to ./models/penguin_01/PenguinLearning.nn
GLOBALS: 'is_continuous_control', 'version_number', 'memory_size', 'action_output_shape'
IN: 'vector_observation': [-1, 1, 1, 33] => 'main_graph_0/hidden_0/BiasAdd'
IN: 'action_masks': [-1, 1, 1, 5] => 'strided_slice_2'
IN: 'action_masks': [-1, 1, 1, 5] => 'strided_slice_3'
OUT: 'action'
DONE: wrote ./models/penguin_01/PenguinLearning.nn file.
INFO:mlagents.trainers:Exported ./models/penguin_01/PenguinLearning.nn file

(ml-agents-0.14) C:\Users\ak12\Documents\Code\MLAgents\ml-agents-0.14.0>
```

Figure 05: Anaconda prompt window: exported .nn file path

INFERENCE

In this section, you'll set up the penguins to perform inference. This will cause your penguins to make decisions using the neural network you trained in the previous section. Once you've set up the neural network in your project, you no longer need Python for intelligent decision making.

- Create a new folder in your Unity project called *NNModels* inside *Assets\Penguin*.
- Find the .nn file that was exported at the end of training.
- In this case, it's *models\penguin_01-0\PenguinLearning.nn*
- Drag the .nn file into the *NNModels* folder in Unity **(Figure 06)**.

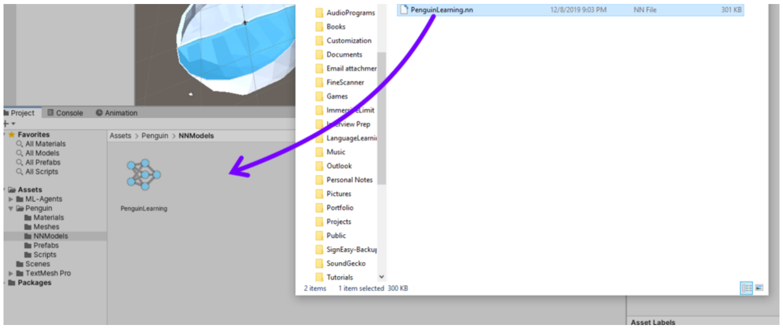
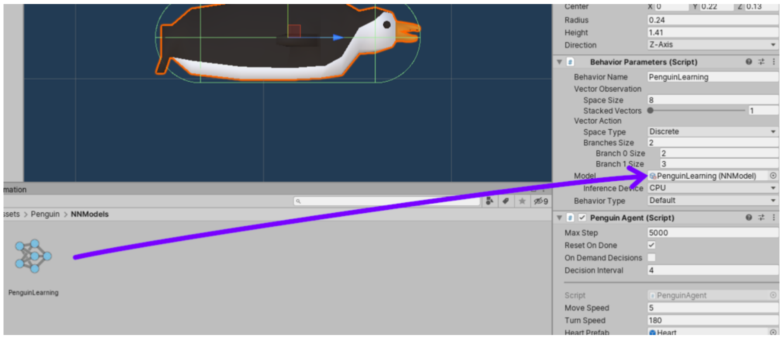


Figure 06: Drag PenguinLearning.nn into the NNModels folder in Unity

- Open the Penguin Prefab by double-clicking it in the Project tab.
- Drag the PenguinLearning nn model into the **Model** field of the **Behavior Parameters** component **(Figure 07)**.


Center X 0 Y 0.22 Z 0.13
Radius 0.24
Height 1.41
Direction Z-Axis

Behavior Parameters (Script)
Behavior Name PenguinLearning
Vector Observation
Space Size 8
Stacked Vectors 1
Vector Action
Space Type Discrete
Branches Size 2
Branch 0 Size 2
Branch 1 Size 3
Model PenguinLearning (NNModel)
Inference Device CPU
Behavior Type Default

Penguin Agent (Script)
Max Step 5000
Reset On Done ☑
On Demand Decisions □
Decision Interval 4
Script PenguinAgent
Move Speed 5
Turn Speed 180
Heart Prefab Heart

Figure 07: Drag the PenguinLearning nn model into the Model field of BehaviorParameters on the Penguin

- Go back to the main Scene and press **Play**.

Penguins - SampleScene - PC, Mac & Linux Standalone - Unity 2019.3.0f1 Personal [PREVIEW PACKAGES IN USE] <DX11>

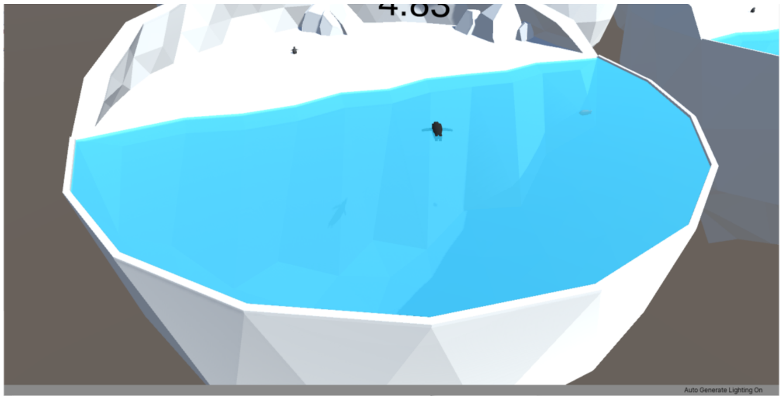


Figure 08: Penguins collecting fish using neural networks to make decisions

The penguins should start catching fish and bringing them to their babies! This is called “inference,” and it means that the neural network is making the decisions. Our agent is giving it observations and reacting to the decisions by taking action.

These .nn files will be included with your game when you build it. They work across all of Unity’s build platforms. If you want to use a neural network in a mobile or console game, you can — no extra installation will be required for your users.

CONCLUSION

You should now have a fully trained ML-Agent penguin that can effectively catch swimming fish and feed them to its baby. Next you should apply what you’ve learned to your own project. Don’t get discouraged if training doesn’t work right away — even this small example required many hours of experimentation to work reliably.

TUTORIAL PARTS

Reinforcement Learning Penguins (Part 1/4)

Reinforcement Learning Penguins (Part 2/4)

Reinforcement Learning Penguins (Part 3/4)

COMMENTS (24)

Most Liked Subscribe via e-mail

Preview Post Comment…

Jon Scott A week ago · 1 Like

Great tutorial, the next step is I want to start using some of this in tandem with arduino controllers.

I have however run into one issue that I am trying to work through.

I start anaconda, run the commands to get everything going.. go over to unity and press play I see maybe 10 to 15 seconds of activity and then I get the following errors.

WARNING:tensorflow:From C:\Users\forum\anaconda3\envs\ml-agents\lib\site-packages\tensorflow_core\python\compat\v2_compat.py:65: disable_resource_variables (from tensorflow.python.ops.variable_scope) is deprecated and will be removed in a future version

```
term
INFO:mlagents_envs:Listening on port 5004. Start
training by pressing the Play button in the Unity
Editor.
INFO:mlagents.trainers:UnityEnvironment worker 0:
environment stopping.
INFO:mlagents.trainers:Learning was interrupted.


Please wait while the graph is generated.
INFO:mlagents.trainers:Saved Model
```

I will keep working to trouble shoot this and if I figure it out I will update this in case anyone else runs into this, but if you have any direction it would be greatly appreciated.

Thanks

Spider Man A week ago · 0 Likes

Hey Scott, were you able to find a solution to this issue, I am also receiving the same error.

Luiz Augusto A day ago · 0 Likes

Clap clap clap!!! Excellent tutorial!! After three days, despite the ups and downs, finally I finished this tutorial. Now, it's time to develop my own agent and environment rules to put in practices my knowledge and experience learned here

This is my first time attempting to set up an ml agent, however when I run the command:

mlagents-learn C:\Users\Chris\Desktop\ML\ml-agents-0.15.1\ml-agents-0.15.1\config\trainer_config.yaml --curriculum C:\Users\Chris\Desktop\ML\ml-agents-0.15.1\ml-agents-0.15.1\config\curricula\ant\ant.yaml --run-id ant_01 --train

The following errors display in the Anaconda Prompt https://imgur.com/a/NFo9zEg

Version Information:

ml-agents: 0.15.1

ml-agents-envs 0.15.1

Communicator API: 0.15.0

TensorFlow: 2.0.1

Python: 3.7.7

Any advice is appreciated, and thank you for the great tutorial so far.

Spider Man A week ago

· 0 Likes

I've resolved my own issue, I missed a couple steps.

Adam Kelly

A week ago · 0 Likes

Glad you got it sorted out, Spider Man

J

Jon Scott A week ago · 0 Likes

throwing an error in my console on unity.

Couldn't connect to trainer on port 5004 using API version API-14. Will perform inference instead.

Adam Kelly A week ago ·

1 Like

Generally this happens when the version of ML-Agents installed in your Anaconda console doesn't match the version in your project. For example, if you install ML-Agents from PyPi instead of the code repo that you used for your project.

J **Jon Scott**

A week ago · 0 Likes

Thanks Adam, I will do some further research and try to verify everything and update my findings here.

J **Jon Scott**

A week ago · 0 Likes

I was able to clear up the error for API Version API-14.. However I still seem to get this error after starting Anaconda, running activate ML agents.. running to press play on unity and that it is listening on 5004.

Once I press play I see the penguins moving as one would expect.. but I never see any periodic updates. It runs for maybe 10 to 15 seconds and then I get this error.

```
INFO:mlagents.trainers:UnityEnvironment worker 0: environment stopping.
INFO:mlagents.trainers:Learning was interrupted. Please wait while the graph is generated.
INFO:mlagents.trainers:Saved Model
```

Is there any other log I can check to see what is interrupting it?

Clinton Bell

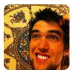
4 days ago · 0 Likes

Jon,

Did you find a solution to this? I'm getting the identical issue.

```
Version information:
ml-agents: 0.14.1,
ml-agents-envs: 0.14.1,
Communicator API: API-14,
TensorFlow: 2.0.1
WARNING:tensorflow:From
```

```
packages\tensorflow_core\python\compat\v2_compat.py:65:
disable_resource_variables
(from
tensorflow.python.ops.variable_scope)
is deprecated and will be
removed in a future version.
Instructions for updating:
non-resource variables are not
supported in the long term
INFO:mlagents_envs:Listening
on port 5004. Start training by
pressing the Play button in the
Unity Editor.
INFO:mlagents.trainers:UnityEnvironment
worker 0: environment stopping.
INFO:mlagents.trainers:Learning
was interrupted. Please wait
while the graph is generated.
INFO:mlagents.trainers:Saved
Model
```



Clinton Bell

4 days ago · 0 Likes

SOLVED:

Amateur move in my case. My objects weren't attached correctly inside the penguin area script GUI. I was getting allowed me to troubleshoot.

Andrew Devonport

A week ago · 0 Likes

Superb, thank you so much for this tutorial.

Far too many you tube videos are out of date as technology moves so fast. I only need to update a few references to get this working with latest ml-agent. The fact you included the tooling setup as well was really appreciated . it has given me something to now explore the subject in more detail.

Keep up the excellent work!

Adam Kelly A week ago · 0 Likes

Thank you very much!

Patris . 4 weeks ago · 0 Likes

Thank you so much for this tutorial, it was very thorough and even with almost no Unity experience I could understand most of it.

Daniel Rodrigues

A month ago · 0 Likes

I get as far as pressing play in the Editor, and

Adam Kelly 3 weeks ago

· 0 Likes

Hey Daniel, there isn't enough to go off of here, but here are a few things to check:

- Is the agent set to have it's behavior type as

"Heuristic" when you are controlling and "Default" when you're training?

- Are you running training? The penguin wont do anything unless (A) you are controlling with the keyboard, (B) you are running training, or (C), you have completed training and have hooked up the .nn model.

David Joiner A month ago · 0 Likes

I'm getting the following error in Unity:

```
Couldn't connect to trainer on port 5004 using API version API-14. Will perform inference instead.
UnityEngine.Debug:Log(Object)
MLAgents.Academy:InitializeEnvironment() (at /Users/joinerd/Desktop/ml-agents-0.14.1/com.unity.ml-agents/Runtime/Academy.cs:286)
MLAgents.Academy:LazyInitialization() (at /Users/joinerd/Desktop/ml-agents-0.14.1/com.unity.ml-agents/Runtime/Academy.cs:161)
MLAgents.Academy:.ctor() (at /Users/joinerd/Desktop/ml-agents-0.14.1/com.unity.ml-agents/Runtime/Academy.cs:150)
MLAgents.<>c:<.ctor>b__50_0() (at
System.Lazy`1:get_Value()
MLAgents.Academy:get_Instance() (at
/Users/joinerd/Desktop/ml-agents-0.14.1/com.unity.ml-agents/Runtime/Academy.cs:63)
MLAgents.Agent:LazyInitialize() (at
/Users/joinerd/Desktop/ml-agents-0.14.1/com.unity.ml-agents/Runtime/Agent.cs:244)
MLAgents.Agent:OnEnable() (at

/Users/joinerd/Desktop/ml-agents-0.14.1/com.unity.ml-agents/Runtime/Agent.cs:201)
```

When I start the learn command, I see the following

```
Version information:
ml-agents: 0.14.1,
ml-agents-envs: 0.14.1,
Communicator API: API-14,
TensorFlow: 2.0.1
INFO:mlagents_envs:Listening on port 5004. Start training by pressing the Play button in the Unity Editor.
```

I've read the problems online with HTTPD_proxy, I do not have any proxies set and I don;t have any proxies showing up if I run print_env

Any thoughts?

Adam Kelly A month ago

· 0 Likes

I've seen this happen when the version of ML-Agents I imported to my Unity project did not match the version I was using with Python. If that's not the case for you, I would ask in the agents.453/

David Joiner

A month ago · 0 Likes

So I wen t back and redid everything from scratch, and it is connecting and

run ning now, but I am getting a weird error

Tag: fis is not defined. UnityEngine.GameObject:CompareTag(String) MLAgents.Sensor.RayPerceptionSensor:PerceiveStatic(Single, IReadOnlyList`1, IReadOnlyList`1, Single, Single, Single, Transform, CastType, Single[], Int32, DebugDisplayInfo) (at /Users/joinerd/Desktop/ml-agents-0.14.0/com.unity.ml-agents/Runtime/Sensor/RayPerceptionSensor.cs:280) MLAgents.Sensor.RayPerceptionSensor:Write(WriteAdapter) (at /Users/joinerd/Desktop/ml-agents-0.14.0/com.unity.ml-agents/Runtime/Sensor/RayPerceptionSensor.cs:102) MLAgents.GrpcExtensions:GetObservationProto(ISensor, WriteAdapter) (at /Users/joinerd/Desktop/ml-agents-0.14.0/com.unity.ml-agents/Runtime/Grpc/GrpcExtensions.cs:217) MLAgents.RpcCommunicator:PutObservations(String, AgentInfo, List`1) (at /Users/joinerd/Desktop/ml-agents-0.14.0/com.unity.ml-`

```
List1) (at /Users/joinerd/Desktop/ml-agents-0.14.0/com.unity.ml-agents/Runtime/Policy/RemotePolicy.cs:34)
MLAgents.Agent:SendInfoToBrain() (at /Users/joinerd/Desktop/ml-agents-0.14.0/com.unity.ml-agents/Runtime/Agent.cs:526)
MLAgents.Agent:SendInfo() (at

/Users/joinerd/Desktop/ml-agents-0.14.0/com.unity.ml-agents/Runtime/Agent.cs:780)
MLAgents.Academy:EnvironmentStep() (at /Users/joinerd/Desktop/ml-agents-0.14.0/com.unity.ml-agents/Runtime/Academy.cs:392)
MLAgents.AcademyFixedUpdateStepper:FixedUpdate() (at /Users/joinerd/Desktop/ml-agents-0.14.0/com.unity.ml-agents/Runtime/Academy.cs:32)
```

that doesn't seem to reference any of the scripts I edited. I can only find one reference to the fish tag in the scripts, and it is set to compare tags against fish not fis, and the tags in the taglist is fish, and the fish prefab is tagged as fish.

David Joiner

A month ago · 0

Likes

and wasn't seeing a typo there

Katten såklart 2 months ago · 0 Likes

mlagents-learn is not recognized as an internal or external command, operable program or batch file.

What am I missing? Picture of Prompt: https://gyazo.com/42fe0afea9999785c69321490c235b31

Katten såklart

2 months ago · 0 Likes

It worked when going via (base) instead of (ml-agents)? I must have missunderstood something

Adam Kelly

2 months ago · 0 Likes

If you went through installation in (base), that makes sense. You could try the installation steps again with the (ml-agents) activated and it should work: https://www.immersivelimit.com/tutorials/ml-agents-python-setup-anaconda

ML-Agents Python Setup with Anaconda

NEXT

Reinforcement Learning Penguins (Part 3/4) | Unity ML-Agents

ABOUT CONNECT