Finding and in Near-photorealistic Environment using Reinforcement Learning

Zhen Zhang
zgzhen@uw.edu

Background

Finding objects in your home is difficult: according to research [1], the average U.S. household has 300,000 things. It will be very helpful if we could automate it.

AI2-THOR [2] is a near-photorealistic 3D environment that simulates real world indoor scenes, based on Unity game engine. It provides Python API for writing AI agents that explores and interacts with this environment.

Reinforcement learning (RL) is an area of machine learning where an agent learns how to behave in a environment to maximize cumulative rewards by performing actions and receiving reward.

Figure 1: Example scene in AI2-THOR

Task

Agent is expected to find two objects (Bowl, Tomato) in a given scene. Agent can perform the following **actions**:

- `MOVE_AHEAD, ROTATE_{LEFT,RIGHT}, LOOK_{UP,DOWN}, FOUND_{BOWL,TOMATO}, DONE`
- **Rewards** consist of:
  - **Continuous reward after picking up one object**
  - Final goal success reward
  - Step penalty

Model

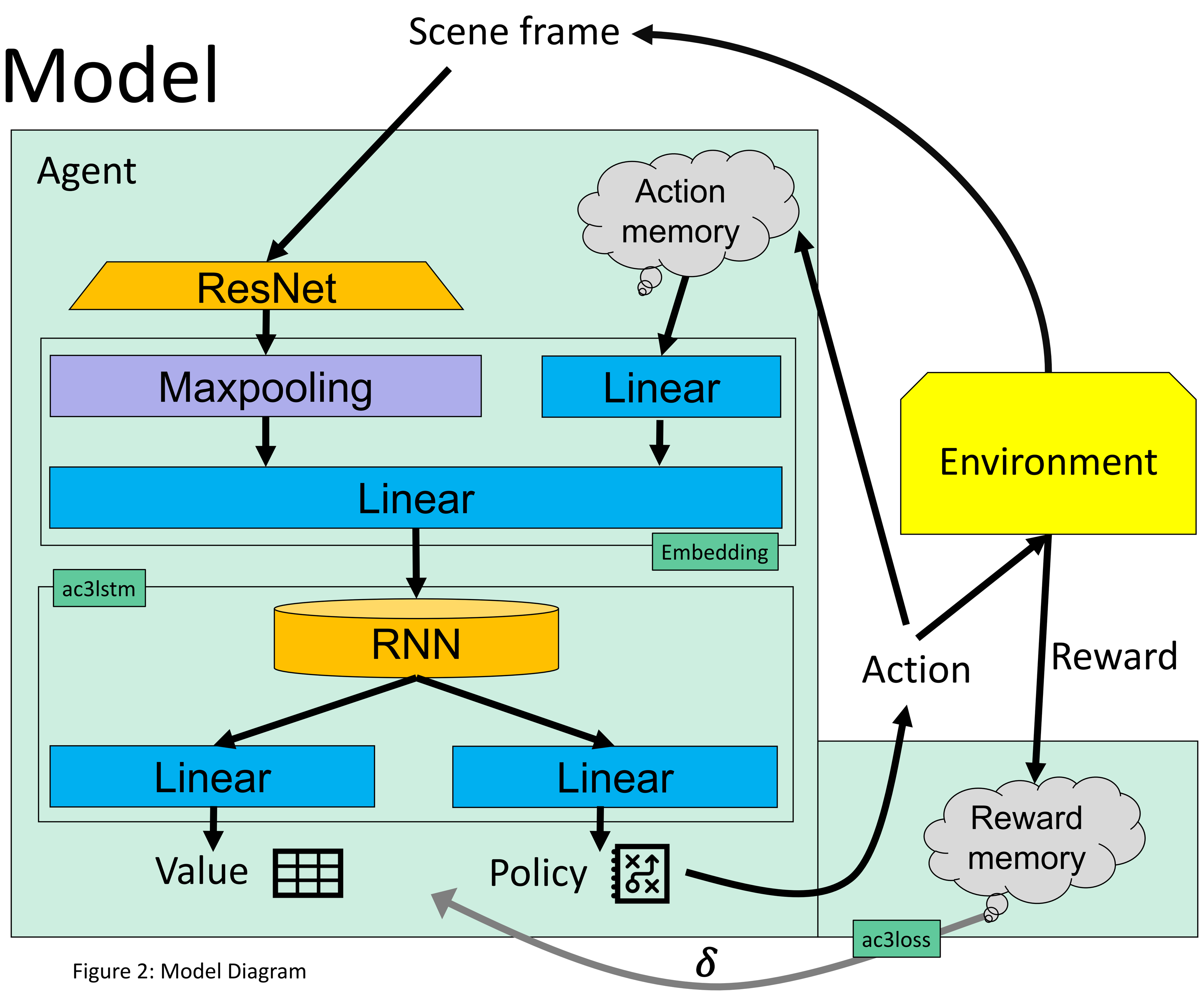

Figure 2: Model Diagram

Lessons learned

1. Understand your model & data. Build effective visualization.
2. Occam's razor: Simplify your problem. Simplify your solution.

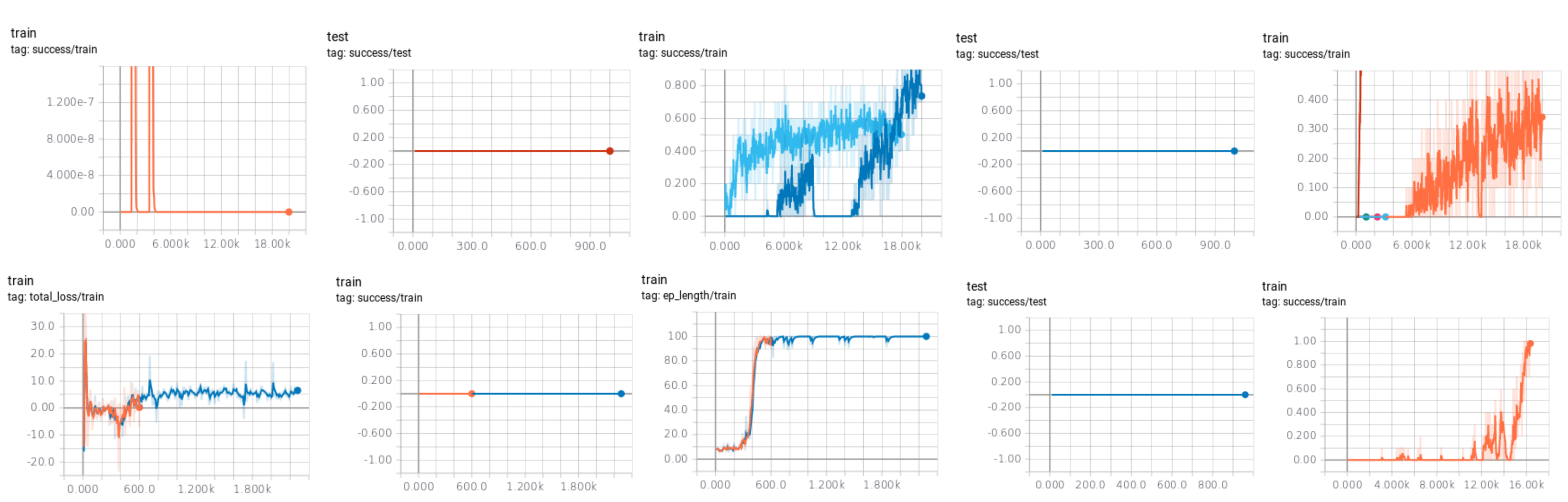

Figure 3: Failed Training in exploration

Research Questions

1. What is a proper size for the hidden layer of augmented state?
2. Continuous rewarding vs. only final lump sum?
3. Should we penalize failed actions and early exits?
4. Will training with more scenes and randomization improve testing results?

Experiments

Setup: GTX 1060 (6GRAM), 6 workers

Baseline configuration:

- Small continuous reward
- No random initialization
- 2 scenes
- Big final reward
- No penalty
- Medium hidden size

	Training steps	Average reward training (testing)	Average success rate training (testing)	Episode length training (testing)
Baseline	18,000	4 (-0.8)	0.5 (0)	30~40 (100)
3 scenes	28,400	2.5 (-0.5)	0.3 (0.01, spikes)	20 (70)
Medium continuous reward	16,000	10 (-1)	0.9 (0)	20 (100)
Penalize failed action	22,000	1.2 (0)	0.2 (0)	15 (15)
Small hidden size	21,000	3 (-0.5)	0.4 (0.04, spikes)	40 (70)
3 scenes, random init, mid cont. reward	21,000	2.0 (**0.4**)	0.1 (**0.06**, more spikes)	40 (60)

Table 1: Results of evaluation

Error Analysis

1. Rotating/Lookup/Lookdown repeatedly in the same place
2. Very short episode – giving up early
3. Find the first object and exit
4. Distracted and not going to the right direction

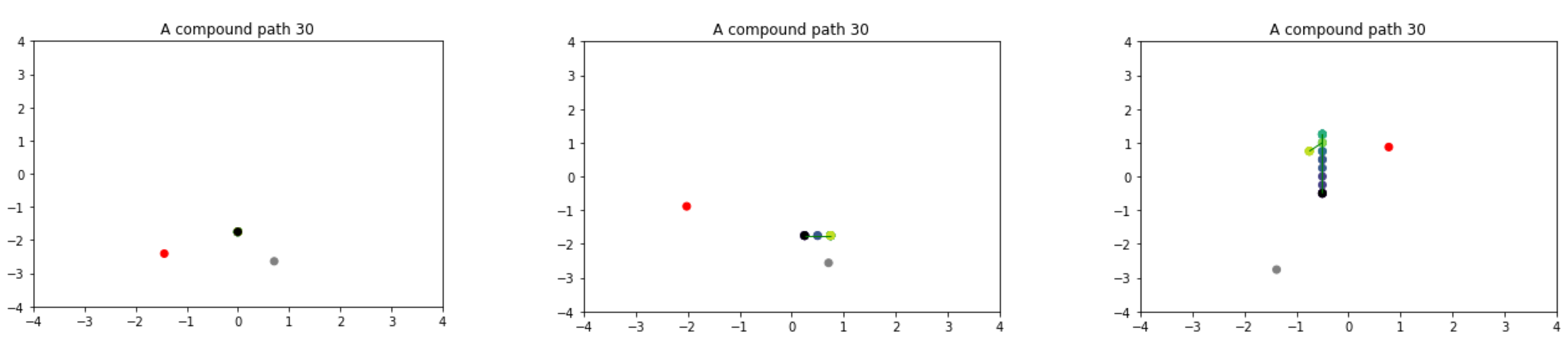

Figure 4: Episode history traces of agent and objects (viewed from top, agent trace is green, tomato is red, bowl is gray)

Conclusion

Reinforcement learning is proven to be useful in the task of finding multiple objects in a simulated environment. However, the test results show that the trained model can't easily generalize to a new scene, or even a new layout of same targets in the same scene. This is similar to the difficulty of human finding objects in unfamiliar environment as well.

[1] http://articles.latimes.com/2014/mar/21/health/la-he-keeping-stuff-20140322
[2] AI2-THOR: An Interactive 3D Environment for Visual AI