Learning to Navigate in Complex Environments

Piotr Mirowski, Razvan Pascanu, Fabio Viola, Hubert Soyer, Andrew J. Ballard, Andrea Banino, Misha Denil, Ross Goroshin, Laurent Sifre, Koray Kavukcuoglu, Dharshan Kumaran, Raia Hadsell

DeepMind, London, UK
Speaker: Li Hao, Lei Zijian

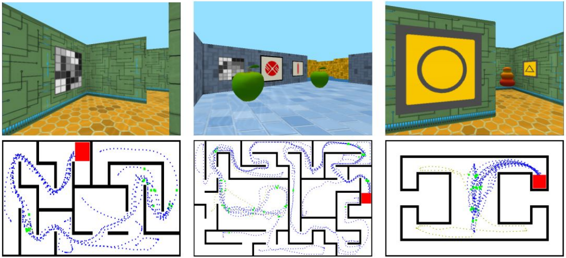
March 12, 2019

Outline

Motivation

Navigational abilities acquisition by policy learning

- In complex environments, navigational abilities could emerge as the by-product of an agent learning a policy that maximizes reward
- Game episode
  - 1, Random start 2, Find the goal (+10)
  - 3, Teleport randomly 4, Re-find the goal (+10) 5, Repeat 3 to 5 (limited time)
  - Fruit reward (apple (+1), strawberry (+2))
- demo: https://youtu.be/INoaTyMZsWI



Motivation

Challenges

- Rewards are often sparsely distributed in the environment
- Environments often comprise dynamic elements, requiring the agent to use memory at different timescales
  - Rapid one-shot memory for the goal location
  - Short term memory for visual observations
  - Longer term memory for constant aspects of the environment (e.g. boundaries, cues)
- These two challenges make the learning process inefficient

Motivation

Accelerate reinforcement learning through auxiliary losses

- Auxiliary tasks have often been used to facilitate representation learning[1]
- In deep RL, auxiliary tasks also works
  - Fit a recurrent model better by predicting next observed state [2]
  - The DQN agent in first-person shooter game is enhanced by an enemy-detection task [3]
- Derive spatial knowledge from auxiliary tasks
  - Depth prediction
  - Local loop closure prediction

---

[1] Suddarth, Steven C. et. al. "Rule-injection hints as a means of improving network performance and learning time." Neural Networks. Springer, 1990.

[2] Li, Xiujun, et al. "Recurrent reinforcement learning: a hybrid approach." arXiv preprint arXiv:1509.03044 (2015).

[3] Lample, et. al. "Playing FPS games with deep reinforcement learning." Thirty-First AAAI Conference on Artificial Intelligence. 2017.

Motivation

Depth prediction and local loop closure prediction

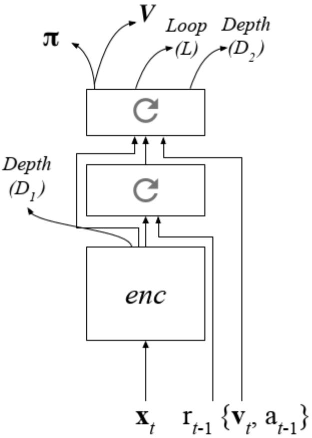
- Depth prediction
  - The depth information might supply valuable information about the 3D structure of the environment.
  - The primary input to the agent is in the form of RGB images
  - A single frame can be enough to predict depth [4]
- Local loop closure prediction
  - Loop closure is to recognize a previously-visited location and update beliefs accordingly
  - Local loop closure is valuable for a navigating agent, since can be used for efficient exploration and spatial reasoning

[4] Eigen, et. al. "Depth map prediction from a single image using a multi-scale deep network." Advances in neural information processing systems. 2014.

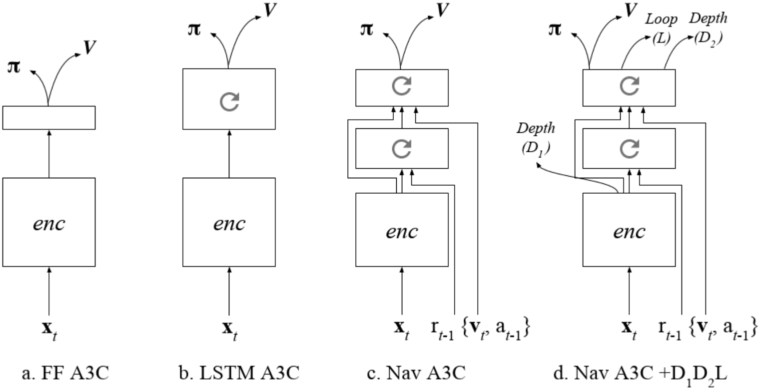
Approach

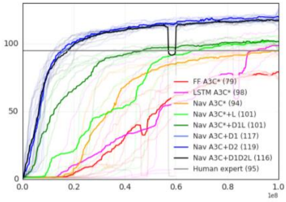
A end-to-end learning framework

- Convolutional encoder and RGB inputs
- Stacked LSTM
- Additional inputs (reward, action and velocity)
- Auxiliary task 1: Depth prediction
- Auxiliary task 2: Loop closure prediction
- The reinforcement learning problem is addressed with A3C algorithm
- Reward clipping is used to stabilize learning: A3C*

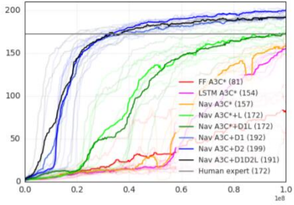


Approach
Variations in architecture

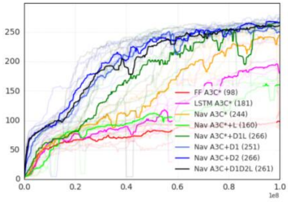




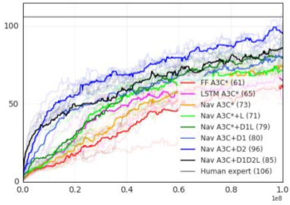
(a) Static maze (small)



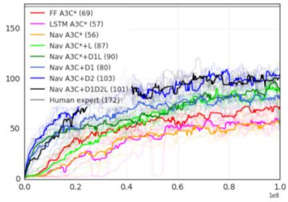
(b) Static maze (large)



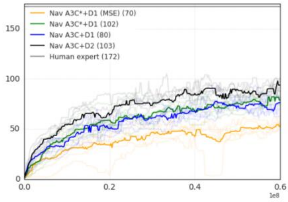
(c) Random Goal I-maze



(d) Random Goal maze (small)



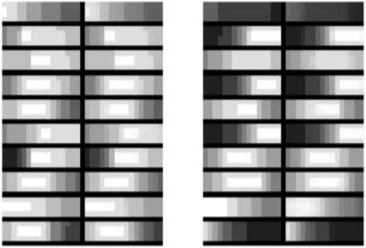
(e) Random Goal maze (large)

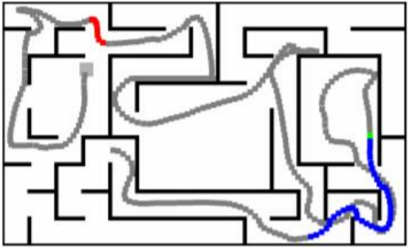


(f) Random Goal maze (large): different formulation of depth prediction

[5] star in the label indicates the use of reward clipping

- Left: Example of depth prediction (pairs of ground truth and predicted depth)
- Right: Example of loop closure prediction





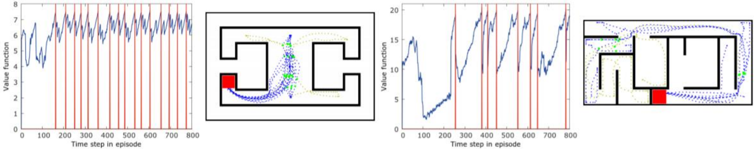
Experiment

Maze	Agent	Mean over top 5 agents AUC	Score	% Human	Highest reward agent Goals	Position Acc	Latency 1:>1	Score
I-Maze	FF A3C*	75.5	98	-	94/100	42.2	9.3s:9.0s	102
	LSTM A3C*	112.4	244	-	**100/100**	**87.8**	15.3s:3.2s	203
	Nav A3C*+$D_1 L$	169.7	266	-	**100/100**	68.5	10.7s:2.7s	252
	Nav A3C+D_2	**203.5**	**268**	-	**100/100**	62.3	**8.8s:2.5s**	**269**
	Nav A3C+$D_1 D_2 L$	199.9	258	-	**100/100**	61.0	**9.9s:2.5s**	251
Static 1	FF A3C*	41.3	79	83	100/100	64.3	8.8s:8.7s	84
	LSTM A3C*	44.3	98	103	100/100	88.6	6.1s:5.9s	110
	Nav A3C+D_2	**104.3**	**119**	**125**	100/100	**95.4**	**5.9s:5.4s**	122
	Nav A3C+$D_1 D_2 L$	102.3	116	122	100/100	94.5	**5.9s:5.4s**	**123**
Static 2	FF A3C*	35.8	81	47	100/100	55.6	24.2s:22.9s	111
	LSTM A3C*	46.0	153	91	100/100	80.4	15.5s:14.9s	155
	Nav A3C+D_2	**157.6**	**200**	**116**	100/100	**94.0**	10.9s:11.0s	**202**
	Nav A3C+$D_1 D_2 L$	156.1	192	112	100/100	92.6	11.1s:12.0s	192
Random Goal 1	FF A3C*	37.5	61	57.5	88/100	51.8	11.0:9.9s	64
	LSTM A3C*	46.6	65	61.3	85/100	51.1	11.1s:9.2s	66
	Nav A3C+D_2	**71.1**	**96**	**91**	**100/100**	**85.5**	**14.0s:7.1s**	**91**
	Nav A3C+$D_1 D_2 L$	64.2	81	76	81/100	83.7	11.5s:7.2s	74.6
Random Goal 2	FF A3C*	50.0	69	40.1	**93/100**	30.0	27.3s:28.2s	77
	LSTM A3C*	37.5	57	32.6	74/100	33.4	21.5s:29.7s	51.3
	Nav A3C*+$D_1 L$	62.5	90	52.3	90/100	51.0	17.9s:18.4s	106
	Nav A3C+D_2	**82.1**	**103**	**59**	79/100	72.4	**15.4s:15.0s**	**109**
	Nav A3C+$D_1 D_2 L$	78.5	91	53	74/100	**81.5**	15.9s:16.0s	102

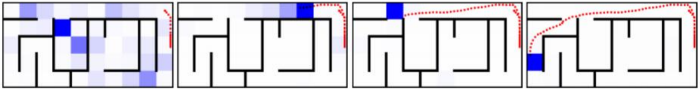
Analysis

Position decoding

- Trajectories of the agent



- Example of position decoding by the Nav A3C+D_2



Analysis

Different combination of auxiliary tasks

- comparison of reward prediction [6] and depth prediction.

Maze	Navigation agent architecture					
	Nav A3C*	Nav A3C+D_1	Nav A3C+D_2	Nav A3C+D_1D_2	Nav A3C*+R	Nav A3C+RD_2
I-Maze	143.3	196.7	**203.5**	197.2	128.2	191.8
Static 1	60.1	103.2	104.3	100.3	86.9	**105.1**
Static 2	59.9	153.1	**157.6**	151.6	100.6	155.5
Random Goal 1	45.5	57.6	71.1	63.2	54.4	**72.3**
Random Goal 2	37.0	66.0	**82.1**	75.1	68.3	80.1

Combining reward prediction and depth prediction (Nav A3C+RD_2) yields comparable results to depth prediction alone (Nav A3C+D_2); normalised average AUC values are respectively 0.995 vs. 0.981.

[6] Jaderberg, et. al. "Reinforcement learning with unsupervised auxiliary tasks",ICLR 2017

Summary

- They proposed a deep RL method,augmented with memory and auxiliary learning target.
- Their approach allows end-end learning, and their auxiliary losses do not rely on any form of replay.
- Whilst their best performing agents are relatively successful at navigation, their abilities would be stretched , due to the limited capacity of the stacked LSTM

Thank You for Your Attention!
Q&A