Learning to communicate with Deep Multi-Agent Reinforcement Learning

Bio-Medical Computing Laboratory

TAEHEUM CHO

03th JUL, 2018

INTRODUCTION

- First, we propose a set of multi-agent benchmark tasks that require communication.
- Second, we formulate several learning algorithms for these tasks.
- Finally, we analyze how these algorithms learn, or fail to learn, communication protocols for the agents.
- The tasks that we consider are fully cooperative, partially observable, sequential multi-agent decision making problems.
- All the agents share the goal of maximizing the same discounted sum of reward.
- Each agent can also communicate with its fellow agents via a discrete limited-bandwidth channel.
- Due to the partial observability and limited channel capacity, the agents must discover a communication protocol that enables them to coordinate their behavior and solve the task.

BACKGROUND

- Deep Q-Network(DQN)
- Independent DQN
- Deep Recurrent Q-Network(DRQN)

Table 1. Rewarding schemes to explore the transition from competitive to the cooperative strategy.

	L player scores	R player scores
L player reward	ρ	-1
R player reward	-1	ρ

For the cases we study $\rho \in [-1, 1]$. Example: with $\rho = -0.5$, when the left player scores, it receives -0.5 points and the right player receives -1 points.

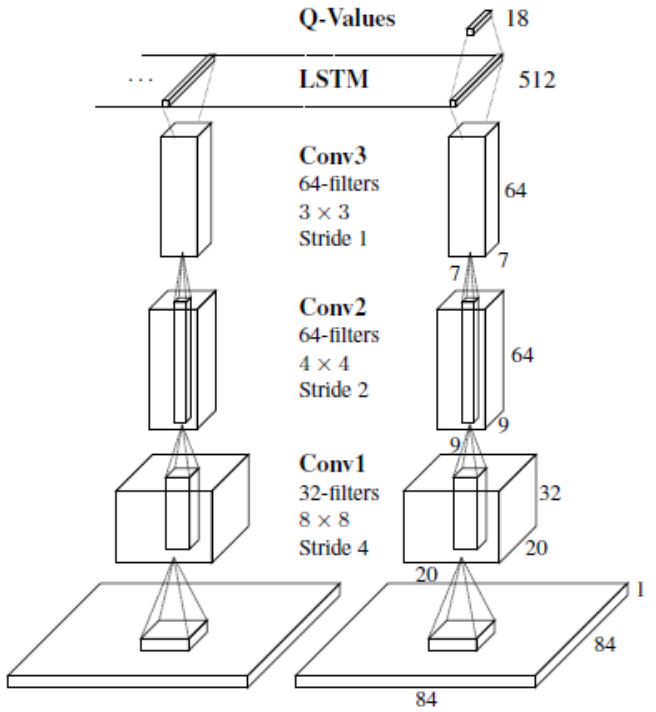


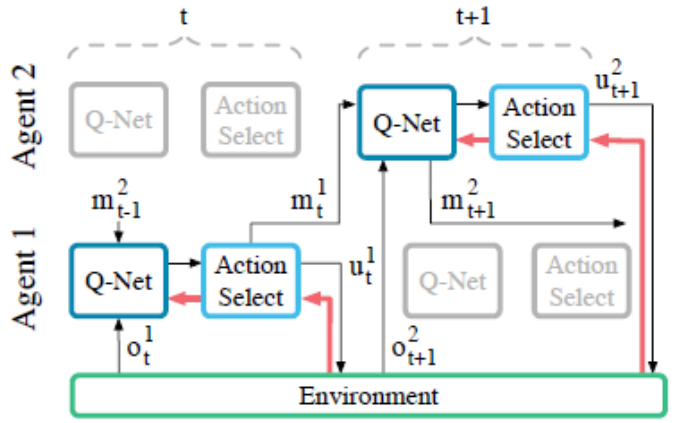
Figure 2: DRQN convolves three times over a single-channel image of the game screen. The resulting activations are processed through time by an LSTM layer. The last two timesteps are shown here. LSTM outputs become Q-Values after passing through a fully-connected layer. Convolutional filters are depicted by rectangular sub-boxes with pointed tops.

SETTING

- All the agents share the goal of maximizing the same discounted sum of rewards R_t.
- While no agent can observe the underlying Markov state s_t, each agent receives a private observation o_t^a correlated to s_t.
- In each time-step, the agents select an environment action $u \in U$ that affects the environment, and a communication action $m \in M$ that is observed by other agents but has no direct impact on the environment or reward.
- As no communication protocol is given a priori, the agents must develop and agree upon such a protocol to solve the task.
- We focus on settings with centralized learning but decentralized execution.

METHOD – Reinforcement Inter-Agent Learning (RIAL)

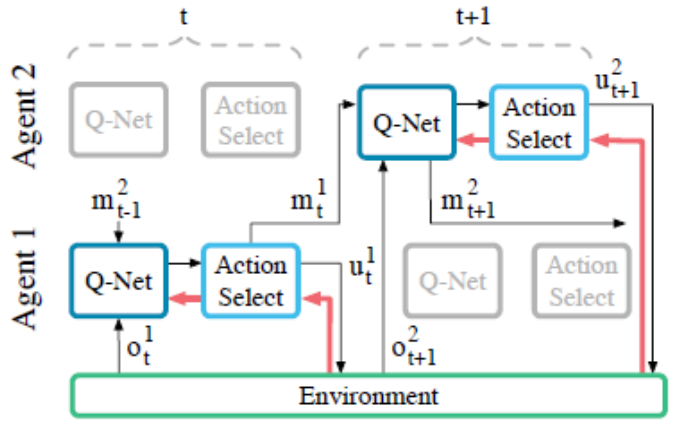
- Reinforced inter-agent learning (RIAL) is to combine DRQN with independent Q-learning for action and communication selection.
- we split the network into Q_u^a and Q_m^a, the Q-values for the environment and communication actions respectively. Similarly, the action selector separately picks u_t^a and m_t^a from Q_u and Q_m, using an ε-greedy policy.
- Figure(a) shows how information flows between agents and the environment, and how Q-values are processed by the action selector in order to produce the action, u_t^a, and message m_t^a.



(a) RIAL - RL based communication

METHOD – Reinforcement Inter-Agent Learning (RIAL)

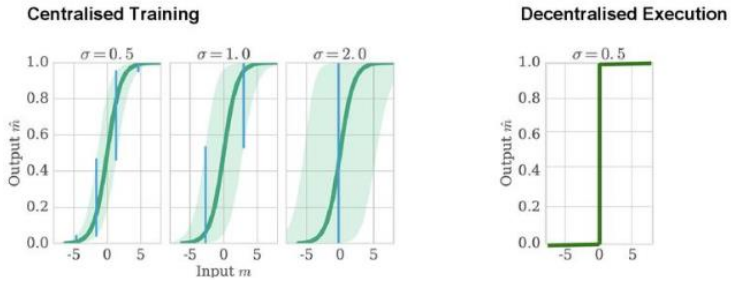
- This variation learns only one network, which is used by all agents. In addition, each agent receives its own index a as input, allowing them to specialize.
- The rich representations in deep Q-networks can facilitate the learning of a common policy while also allowing for specialization.
- Parameter sharing also dramatically reduces the number of parameters that must be learned, thereby speeding learning.



(a) RIAL - RL based communication

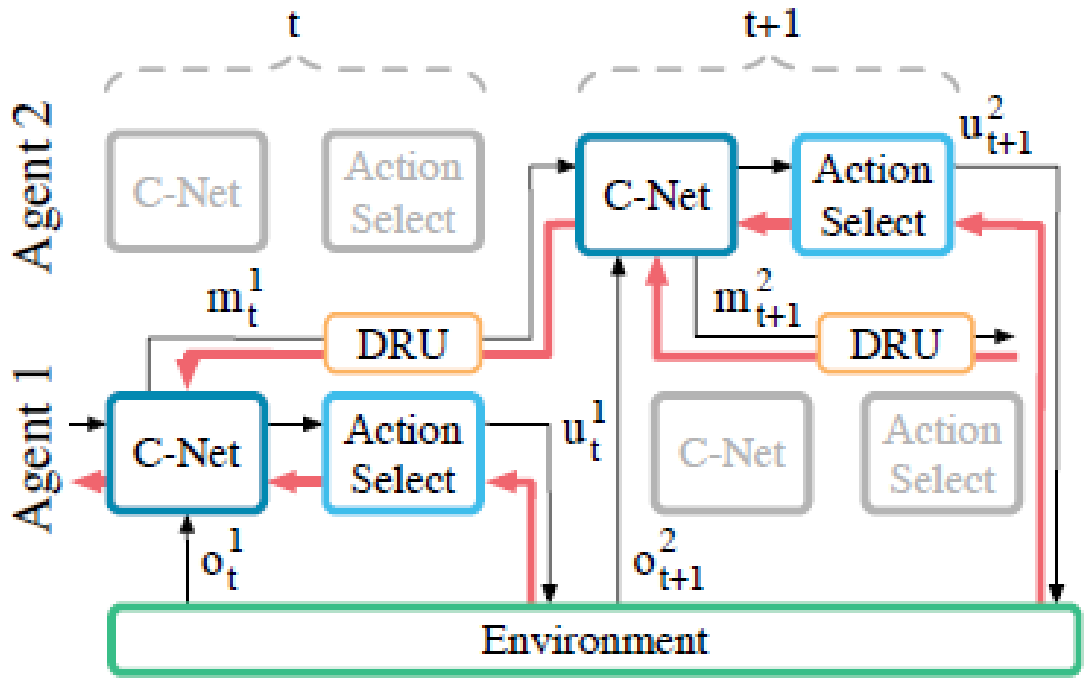
METHOD – Differentiable Inter-Agent Learning (DIAL)

- The agents do not give each other feedback about their communication actions in RIAL.
- The main insight behind DIAL is that the combination of centralised learning and Q-networks makes it possible, not only to share parameters but to push gradients from one agent to another through the communication channel.
- Thus, while RIAL is end-to-end trainable within each agent, DIAL is end-to-end trainable across agents.
- C-Net outputs two distinct types of values as shown in Figure(b)
  - $Q(\cdot)$, the Q-values for the environment actions, which are fed to the action selector.
  - m_t^a, the real-valued message to other agents, which bypasses the action selector and is instead processed by the discretize/regularize unit (DRU(m_t^a)).



METHOD – Differentiable Inter-Agent Learning (DIAL)

- In DIAL, the gradient term for m is the backpropagated error from the recipient of the message to the sender.

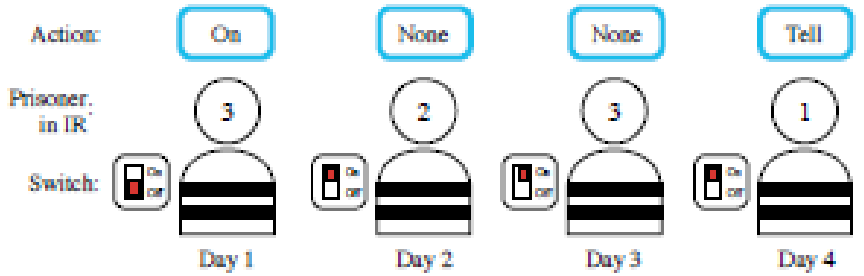


(b) DIAL - Differentiable communication

EXPERIMENT – Switch Riddle

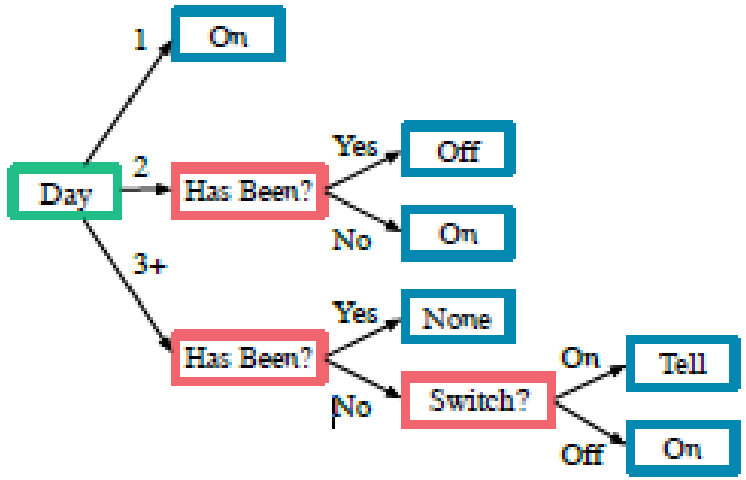
- *One hundred prisoners have been newly ushered into prison. The warden tells them that starting tomorrow, each of them will be placed in an isolated cell, unable to communicate amongst each other. Each day, the warden will choose one of the prisoners uniformly at random with replacement, and place him in a central interrogation room containing only a light bulb with a toggle switch. The prisoner will be able to observe the current state of the light bulb. If he wishes, he can toggle the light bulb. He also has the option of announcing that he believes all prisoners have visited the interrogation room at some point in time. If this announcement is true, then all prisoners are set free, but if it is false, all prisoners are executed. The warden leaves and the prisoners huddle together to discuss their fate. Can they agree on a protocol that will guarantee their freedom?*

- 죄수가 한 명씩 랜덤으로 방에 들어가서 스위치를 올렸다 내렸다 할 수 있고 모든 죄수가 왔다 갔는지 생각하여 선언할 수 있다. 거짓이면 모두 처형, 사실이면 모두 사면하는 게임

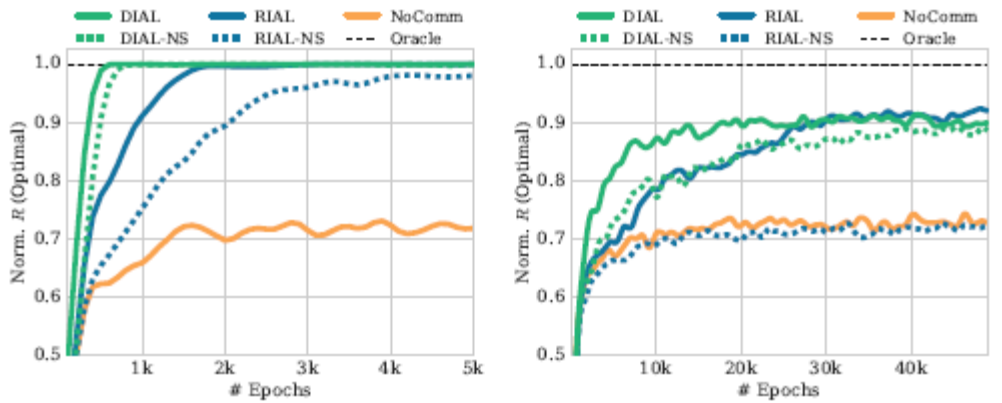


EXPERIMENT – Switch Riddle

- The decision tree extracted for n = 3 to interpret the communication protocol discovered by DIAL.
- Figure(c) is the protocol of the switch riddle game. 즉, 게임의 규칙성을 찾아내는 것



(c) Protocol of $n = 3$



(a) Evaluation of $n = 3$ (b) Evaluation of $n = 4$

NS – Non Parameter sharing
NoComm – no communication
Oracle – Ground Truth

EXPERIMENT – Mnist Game

- 여기서 행동 a는 바이너리 액션을 뜻한다. {0,1}
- a로 맞추는 것이 목적이 아니라... 상대방의 컬러-디짓을 알아내어 내 u(a)를 선택함으로 인해서 가장 큰 reward를 얻는 것이 목적이다.
- 동시에 두개의 게임을 진행한다.

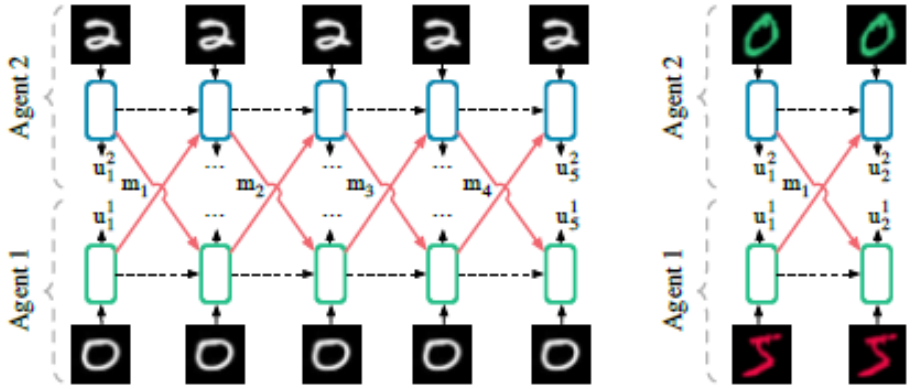


Figure 5: MNIST games architectures.

The reward for each agent is $r(a) = 2(-1)^{a_2^a + c^a + d^{a'}} + (-1)^{a_2^a + d^a + c^{a'}}$

total cooperative reward is $r_2 = r(1) + r(2)$

EXPERIMENT – Mnist Game

- Colour-Digit MNIST
  - each agent observes the pixel values of a random MNIST digit in red or green of size 2*28*28.
  - Only one bit of information can be sent, so agents must agree to encode/decode either colour or parity, with parity yielding greater rewards.
  - The game has two steps; in the first step, both agents send a 1-bit message, in the second step they select a binary action u_2^a.

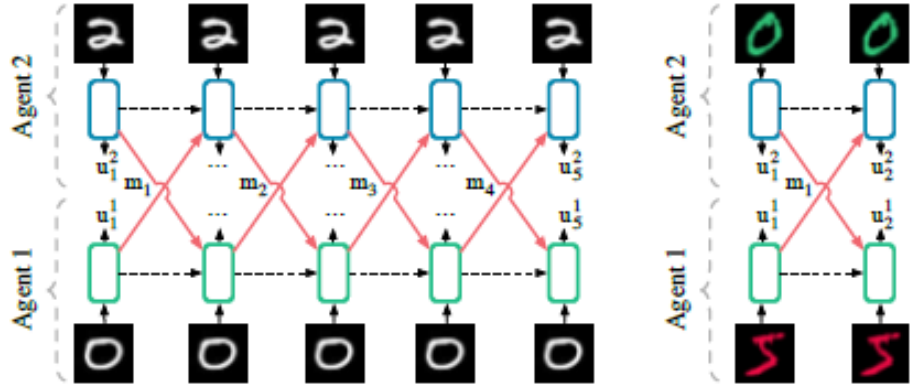
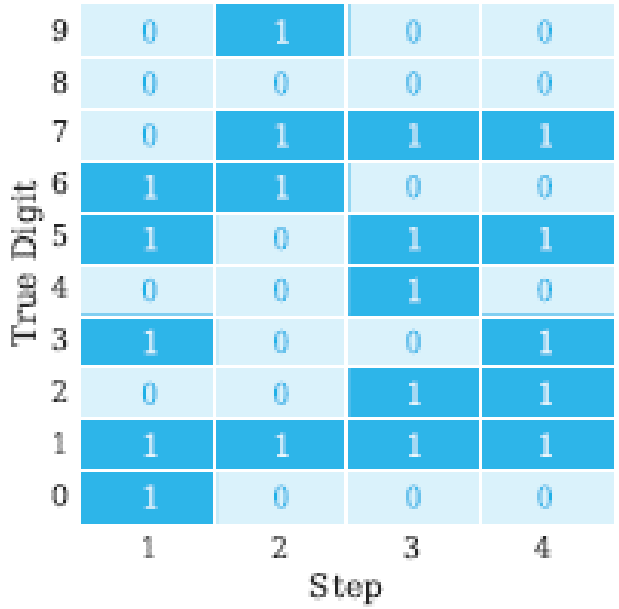


Figure 5: MNIST games architectures.

EXPERIMENT – Mnist Game

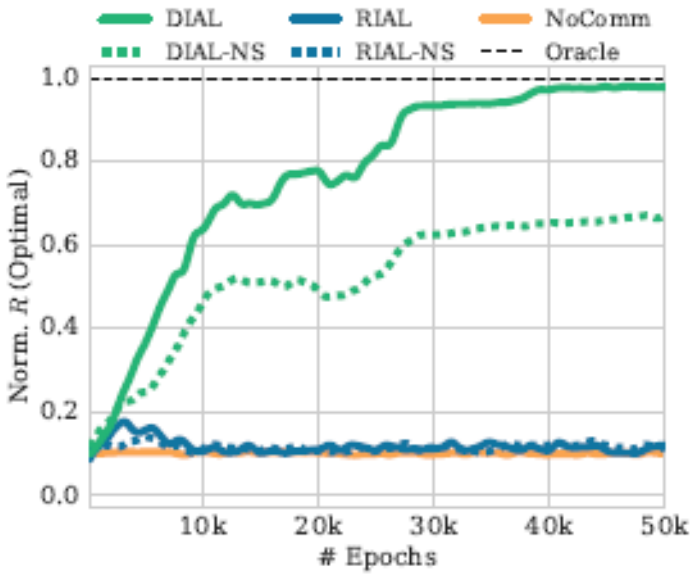
- Multi-Step MNIST
  - Each step the agents send a message, m_t^a, and take an action $u_t^a \in \{0, \ldots, 9\}$.
  - Only at the final step, $t = 5$, is reward given, $r_5 = 0.5$ for each correctly guessed digit, $u_5^a = d^{a'}$.
  - Agents must find a protocol that integrates information across the four messages they exchange (the last message is not received).



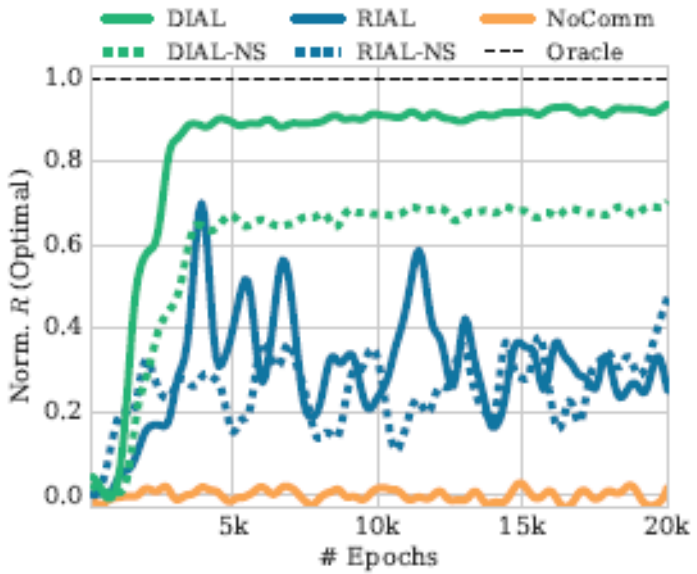
(c) Protocol of Multi-Step

EXPERIMENT – Experimental results

- DIAL substantially outperforms the other methods on both games. Furthermore, parameter sharing is crucial for reaching the optimal protocol.
- DIAL can also optimize the message content with respect to rewards taking place many time-steps later, due to the gradient passing between agents, leading to optimal performance in multi-step MNIST.



(a) Evaluation of Multi-Step



(b) Evaluation of Colour-Digit