COPS RL reading grp

Topic: Communication in MARL systems

2nd Meet

Wednesday: 3rd Feb

Topics covered:

- Emergence of Linguistic Communication From Referential Games with Symbolic and Pixel Input by Angeliki Lazaridou, Karl Moritz Hermann, Karl Tuyls, Stephen Clark (THARUN)
- EMERGENCE OF LANGUAGE WITH MULTI-AGENT GAMES: LEARNING TO COMMUNICATE WITH SEQUENCES OF SYMBOLS by Serhii Havrylov, Ivan Titov. ICLR Workshop, 2017. (YASH)
- Emergence of Grounded Compositional Language in Multi-Agent Populations by Igor Mordatch, Pieter Abbeel. arXiv, 2017. [Post] (VIKHYATH)
- Cooperation and communication in multiagent deep reinforcement learning by Hausknecht M J. 2017. (SOMNATH)
- Learning to communicate to solve riddles with deep distributed recurrent q-networks by Foerster J N, Assael Y M, de Freitas N, et al. arXiv, 2016. (AYUSH)
- Learning to communicate with deep multi-agent reinforcement learning by Foerster J, Assael Y M, de Freitas N, et al. NIPS, 2016. (NISHANT in previous meeting)
- Learning multiagent communication with backpropagation by Sukhbaatar S, Fergus R. NIPS, 2016. (SHRAVAN)

Yash

Emergence of Language from Multi-Agent Games: Learning to Communicate via Sequence of Symbols

Referential Games

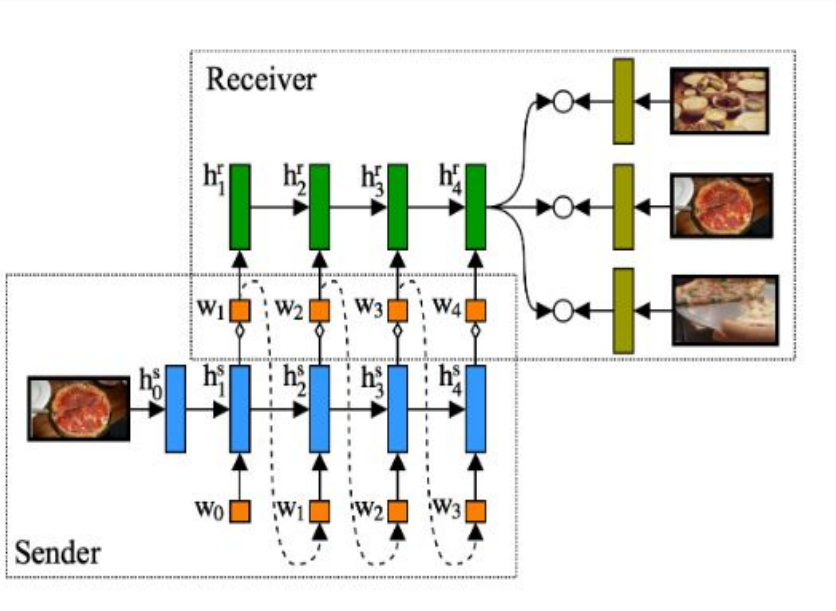
- 2 agents trying to establish communication.
- The sender sees a target image, and send a message to the receiver.
- This message is represented by a sequence of symbols.
- The receiver has many images, where 1 of them is the target image, rest are distracting images.
- The task of the receiver is to identify the target image, based on the message he gets from the receiver.

$$\{i_n\}_{n=1}^{N}$$

$$\{d_k\}_{k=1}^{K}.$$

Agent Architectures

- Both sender and receiver networks are LSTMs
- The sender acts as a language model, by sampling from categorical distributions. Input is the target image, and a start token <S>
- The receiver gets the message and images as input



Loss function

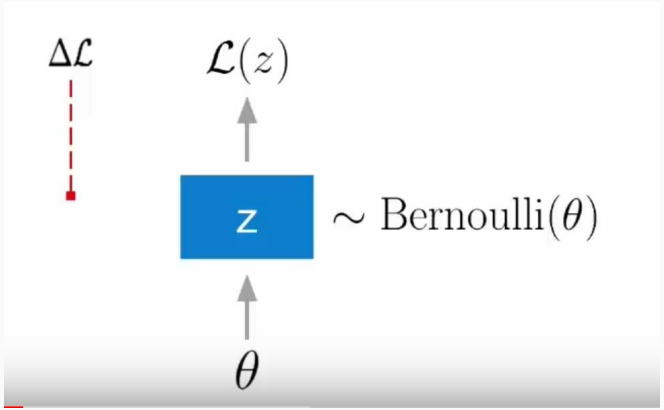
- g(.) is an affine transformation of the images
- g(.) * hidden state is an energy function, which represents the probability distribution over a set of images, should be high for target image
- Also used KL divergence so that the learned protocol is as close as possible to natural language

$$\mathcal{L}_{\phi,\theta}(t) = \sum_{k=1}^{K} \max[0, 1 + g(t)^T h_l^r - g(d_k)^T h_l^r]$$

Gumbel Softmax

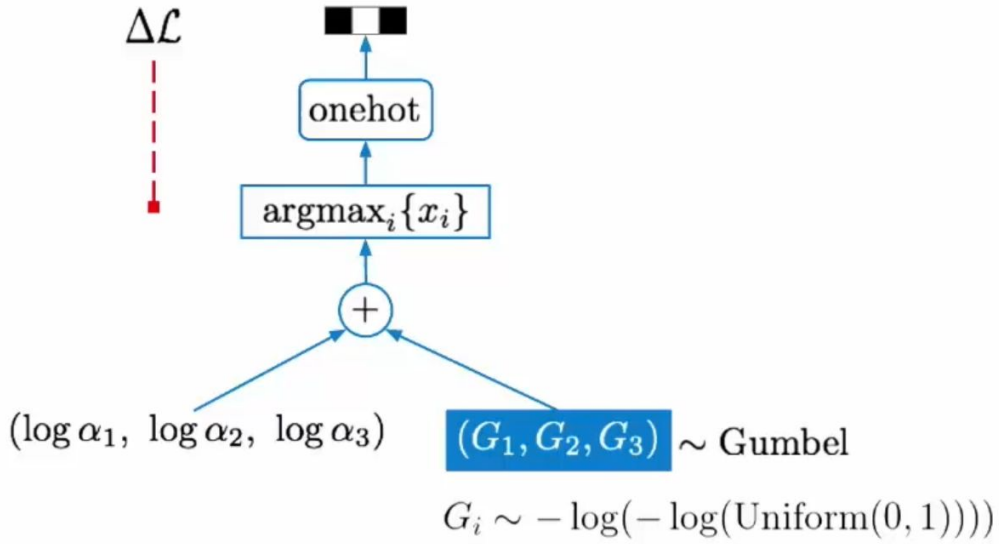
Like in VAE, we have the reparameterization technique, for sampling from a gaussian distribution, we need something for language models.

The difference here is that Gaussian was continuous, but here we have discrete random variables.

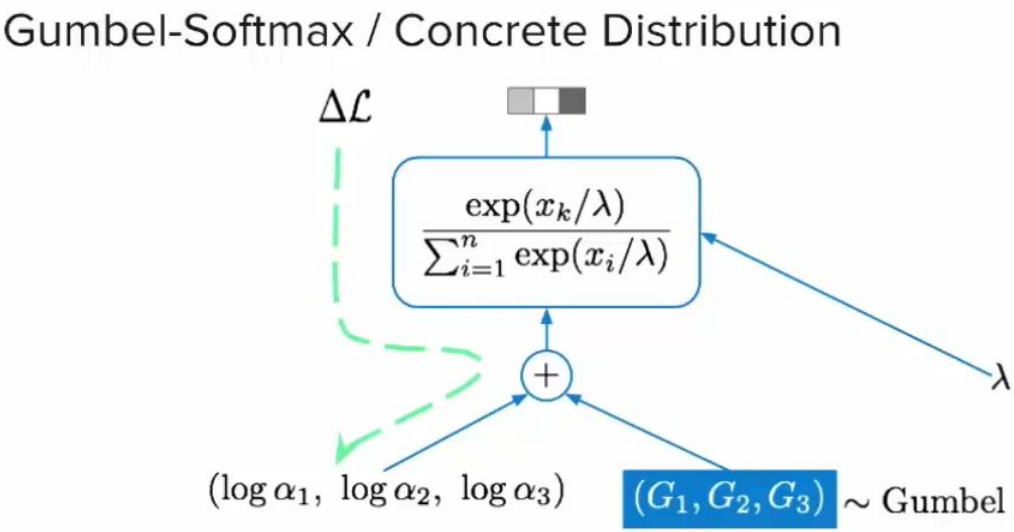


Gumbel Softmax

Gumbel-Max Trick for sampling categoricals



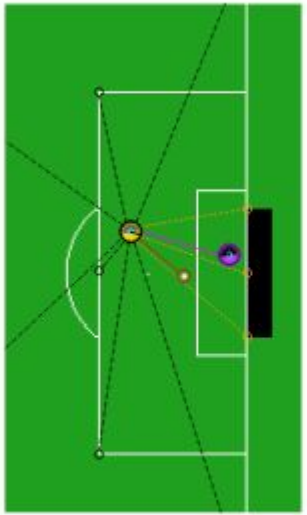
Gumbel Softmax



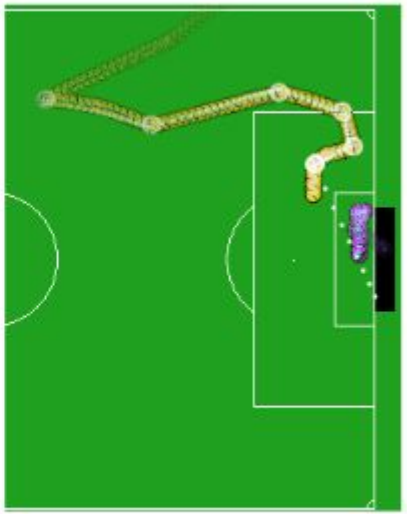
Somnath : Survey of a Cooperative Multi Agent Problem

Learning communication protocols for cooperation.

HFO Half Field Offence



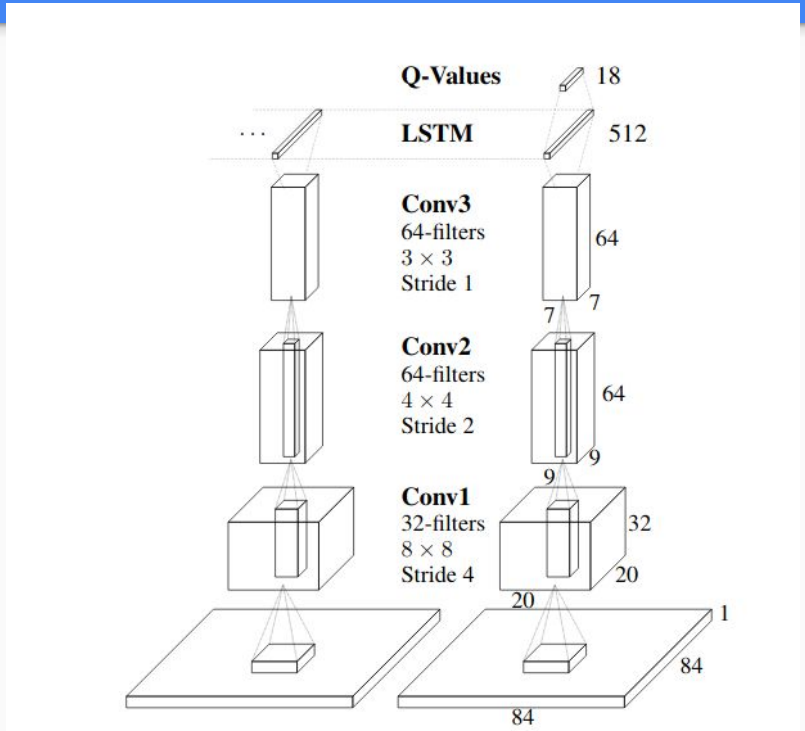
(a) State Space



(b) Helios Champion

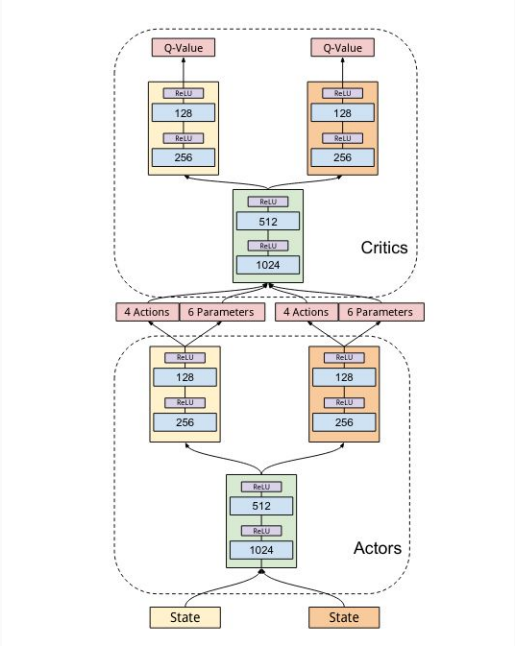
Partial Observable MDP

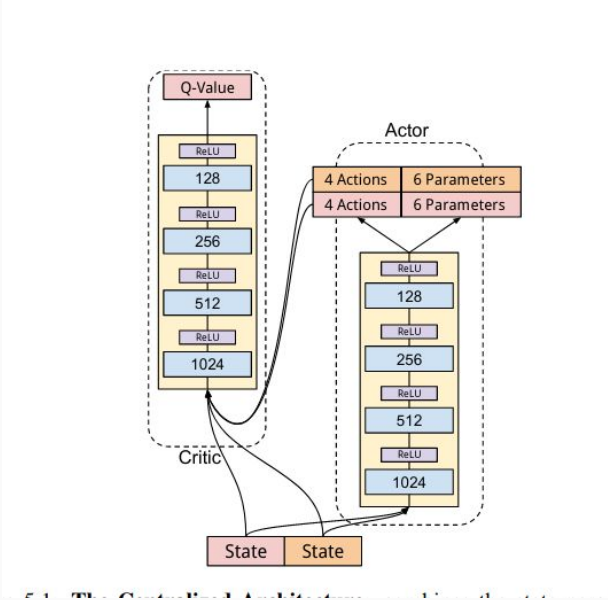
As explained in the previous meet, We use Recurrent Neural Networks to solve this problem, And also other centralized approaches.



Ways of Formulating a MultiAgent Coop.

1. Independent Learning (Baseline)
2. Centralized Control :- One Model n outputs.
3. Parameter Sharing.
4. Memory Sharing : - We share experiences of other agents with each other.





Enjoy!!.

- Independent learning : -
  https://www.cs.utexas.edu/~larg/hausknecht_thesis/2v0_joint.mp4
  One learnt How to goal the other was dummy.
- Centralized Controller : -
  http://www.cs.utexas.edu/~larg/hausknecht_thesis/centralized_2v0.mp4
  Either one started working.
- Parameter Sharing : -
  http://www.cs.utexas.edu/~larg/hausknecht_thesis/shareparams_2layer.mp4
  Both assigned a fixed role in all the episodes
- Replay Memory
  http://www.cs.utexas.edu/~larg/hausknecht_thesis/sharereplay_2v0.mp4
  Both learned similar policies and used to change each others role when needed.

Goal vs Keeper

The above was a simple task and didn’t need a lot of cooperation but the following forces them to learn cooperation by passing the ball.

- Centralized Control
- Parameter Sharing
- Memory Sharing

Importance of Communication.

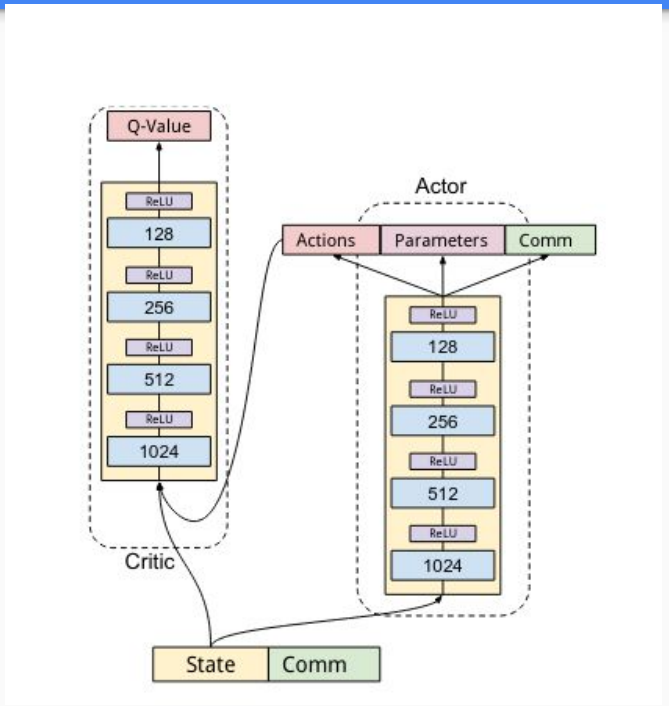
Simpler task can be formed and made into cooperating just by restricting using Reward function or architecture for complex tasks we use communication, Some info to be shared to others. Few explained in the survey

- Independent Communication (Baseline)
- Teammate Communication Gradients
- Grounded Semantic Network

Independent Communication

It’s more like choosing a message with the action vector.

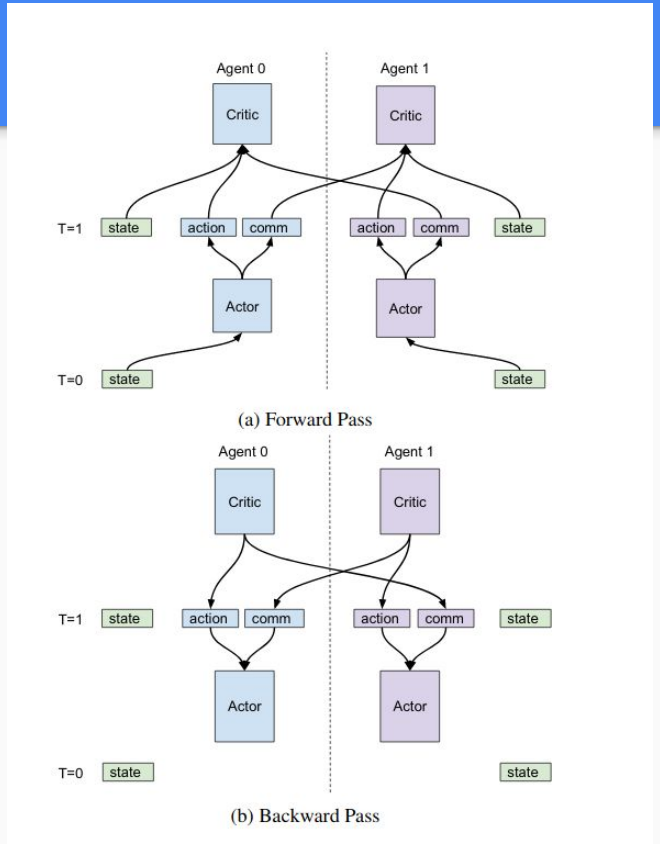
Baseline for update from the same Critic is obtained (independent).



Teammate Communication Gradient

Here we cross over the baseline hence we obtain gradients which has more meaning as the message is concatenated to the state vector of the other agent.

This suffers from nonstationarity of the other agent.



Grounded Semantic Network

The following is more like GANs on metrifing the messaging and generating more meaningful reward.

r(2) is the single step reward of the other agent for the message m(1)

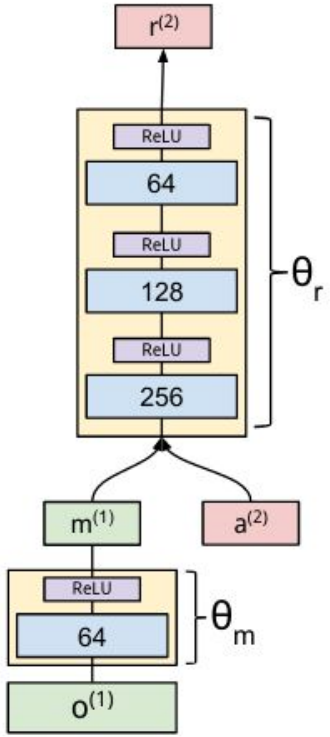
No use of a external Critic network.

Hence its totally unaware of the other agents policy hence can't provide a meaningful message all the time.

$$\hat{r}^{(2)} = R\Big(M\big(o^{(1)};\theta_m\big), a^{(2)};\theta_r\Big) \tag{6.1}$$

GSN training follows a supervised learning paradigm. Given experience tuple $(o^{(1)}, a^{(2)}, r^{(2)})$, the GSN is trained to regress its predictions towards the rewards of the teammate, minimizing the following loss function:

$$L(\theta_r, \theta_m) = \mathbb{E}_{(o^{(1)},a^{(2)},r^{(2)})}\left[\Big(r^{(2)} - R\big(M(o^{(1)};\theta_m), a^{(2)};\theta_r\big)\Big)^2\right] \tag{6.2}$$



Blind move task

- With GSN it was able to learn but only after 10 x more iteration this shows the lack of knowledge the others agent policy.
- The Teammate Gradient fails because it gets a gradient which wants to maximize the reward (because critic does that). But this is something what the agent wants to hear (that is the goal is just forward to it and it can goal (greedy)).

t-SNE projection

The following methods show good performance but are limited in many ways hence many more communication protocols like DIAL etc..

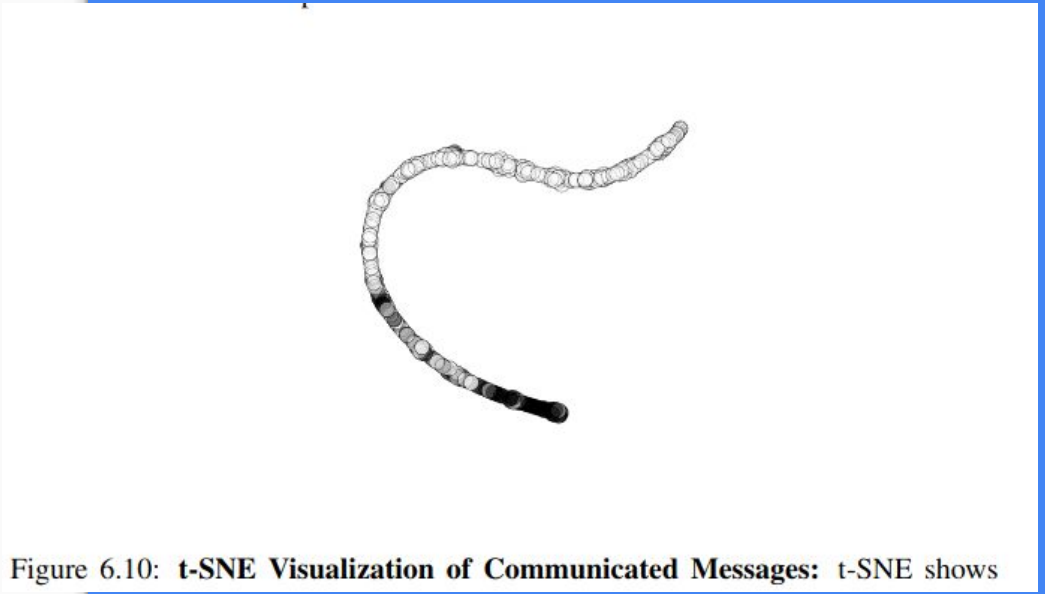


Figure 6.10: **t-SNE Visualization of Communicated Messages:** t-SNE shows

More Related works:
https://repositories.lib.utexas.edu/handle/2152/45681

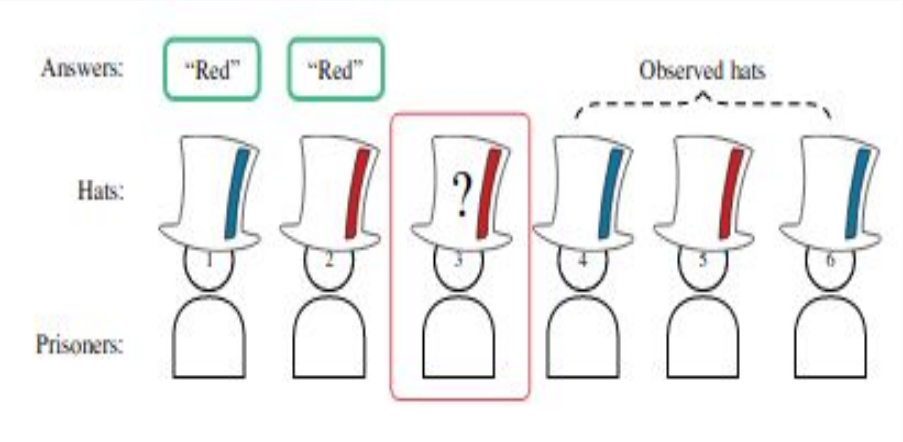
Ayush

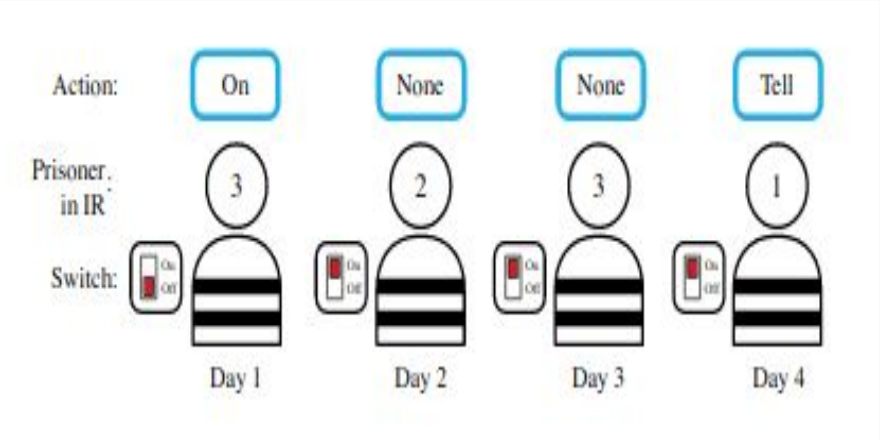
Deep Distributed Recurrent Q Networks to solve multi agent riddles

Comm. in Partially Observable settings

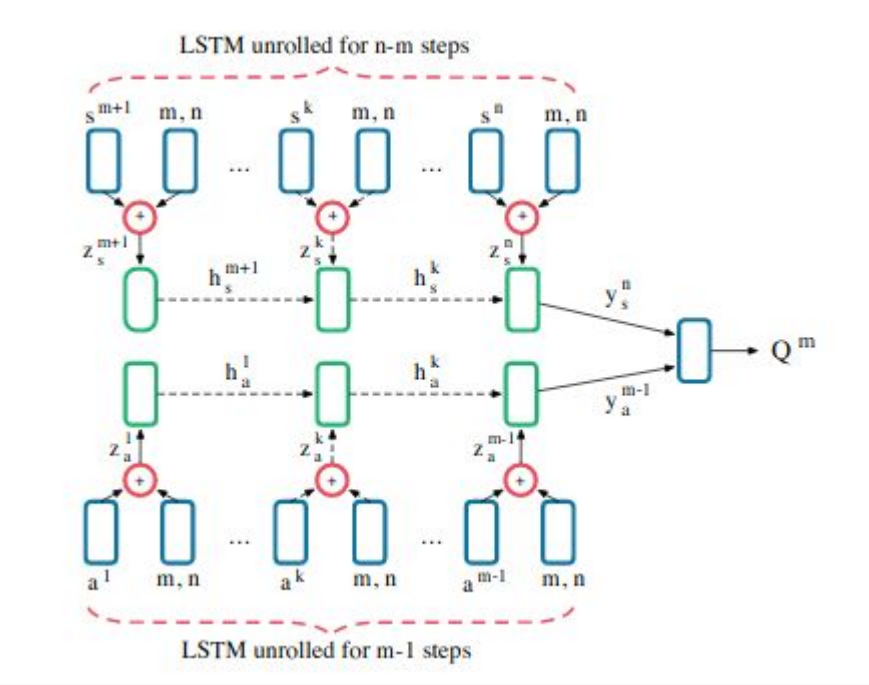
- Agents come up with a communication protocol themselves to cooperate effectively and ultimately get better rewards
- Key Differences from a DQN here are:
  - Last Action Inputs
  - Inter agent weight sharing
  - No Experience replay buffer

Classical Riddles Solved using MARL





cont.


LSTM unrolled for n-m steps
s m+1
m, n
s k
m, n
s n
m, n
z s m+1
z s k
z s n
h s m+1
h s k
y s n
Q m
h a l
h a k
y a m-1
z a l
z a k
z a m-1
a l
m, n
a k
m, n
a m-1
m, n
LSTM unrolled for m-1 steps

Shravan: CommNet

MS Paint!

Tharun

EMERGENCE OF LINGUISTIC COMMUNICATION FROM REFERENTIAL GAMES WITH SYMBOLIC AND PIXEL INPUT

How environmental or pre-linguistic conditions affect the nature of the communication protocol that an agent learns

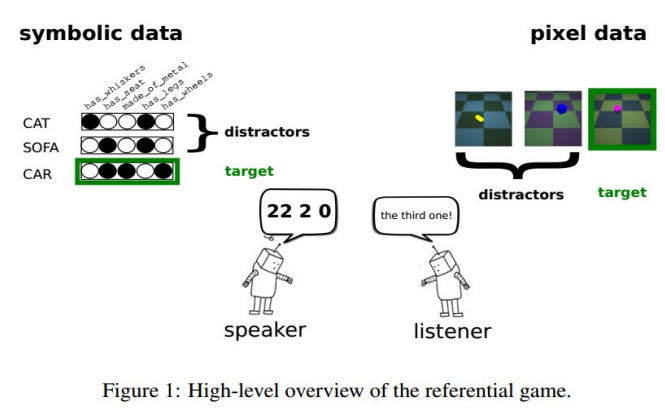


Figure 1: High-level overview of the referential game.

max length	alphabet size	lexicon size	training accuracy
2	10	31	92.0%
5	17	293	98.2%
10	40	355	98.5%

Data	length 2 lexicon size	length 2 acc.	length 5 lexicon size	length 5 acc.	length 10 lexicon size	length 10 acc.
training data	31	92.0	293	98.2	355	98.5
test data	1	74.2	70	76.8	98	81.6
unigram chimera	5	39.3	88	40.5	99	47.0
uniform chimera	3	31.2	87	32.2	100	42.6

Topographical similarity

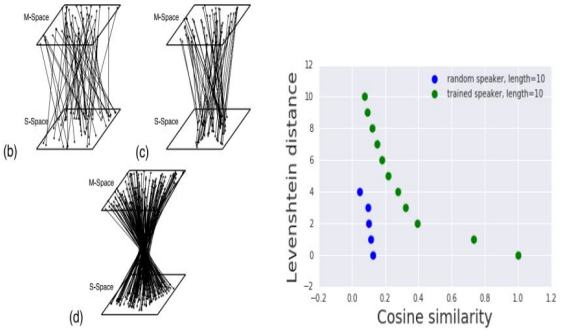


Figure 3: **left**: Three languages with different properties, taken from Brighton & Kirby (2006). The mapping between states and signals shown in (b) is random; there is no relationship between points in the meaning and signal space. In (c) and (d), similar meanings map to similar signals, i.e., there is a topographic relation between meanings and signals. **right**: Relation between objects' cosine similarity and their message Levenshtein distance for trained and random agents.

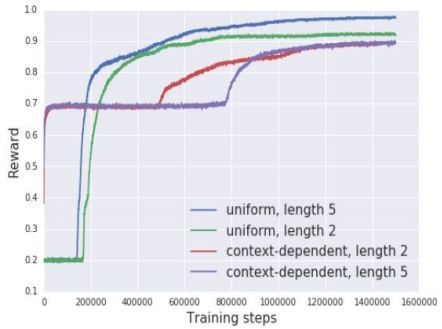


Figure 2: Training curves of different experimental setups with uniform and context-dependent target selection.

Probe models

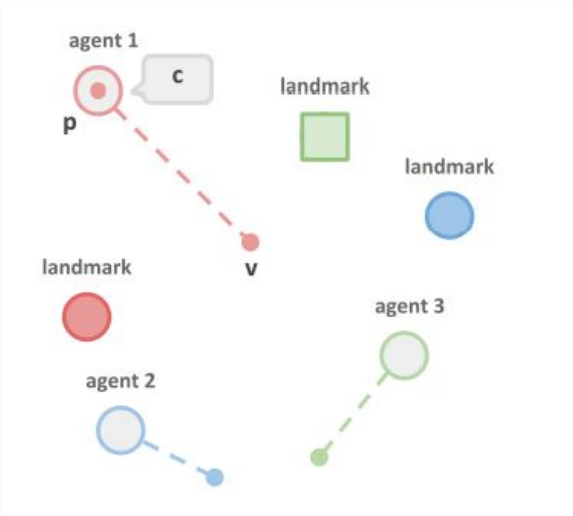
game (random baseline)	**object position** (20.0)	**object shape** (20.0)	**object color** (12.0)	**floor color** (33.0)
A	95.3	90.2	24.7	36.4
B	88.6	41.2	63.8	45.4
C	85.9	43.5	65.8	43.8
D	89.4	47.1	82.0	42.3

Vikhyath

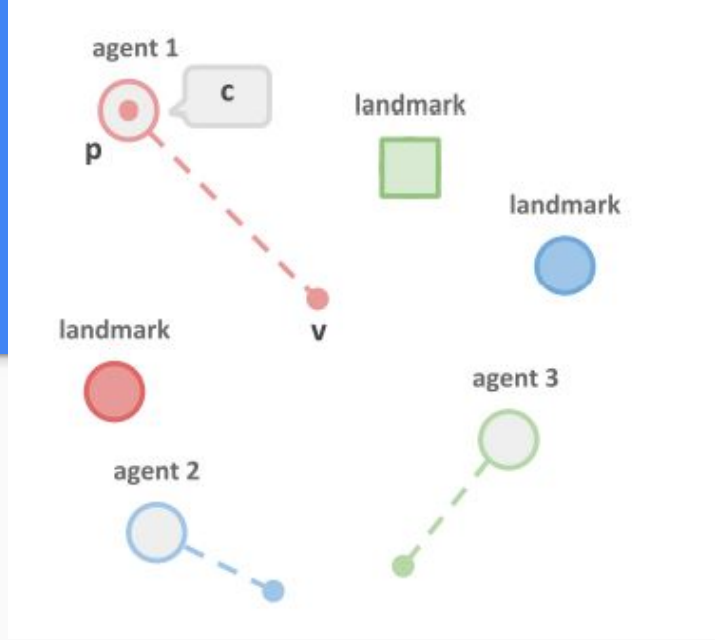
EMERGENCE OF GROUNDED COMPOSITIONAL LANGUAGE IN MULTI-AGENT POPULATIONS

Introduction

- Communication as a necessity
- Abstract symbols for which agents have to give meanings
- Grounded
- Compositional



Environment



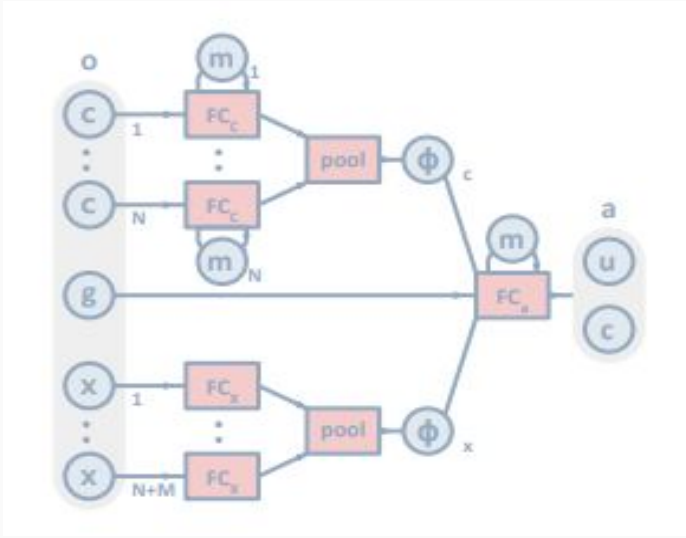
- Each agent can move around.
- At every timestep, agent utters *c* to all agents.
- Each agent has internal goals(g), not visible
- Each agent has a recurrent memory bank, which has no pre-designed behavior
- Observation o = $\left[{}_i\mathbf{x}_{1,\ldots,(N+M)}\ \mathbf{c}_{1,\ldots,N}\ \mathbf{m}_i\ \mathbf{g}_i \right]$.

Policy

- No Q Networks
- No Policy Gradients
- Backprop thru time
- Categorical Communications thru Gumbel-Softmax Estimator

Policy Architecture

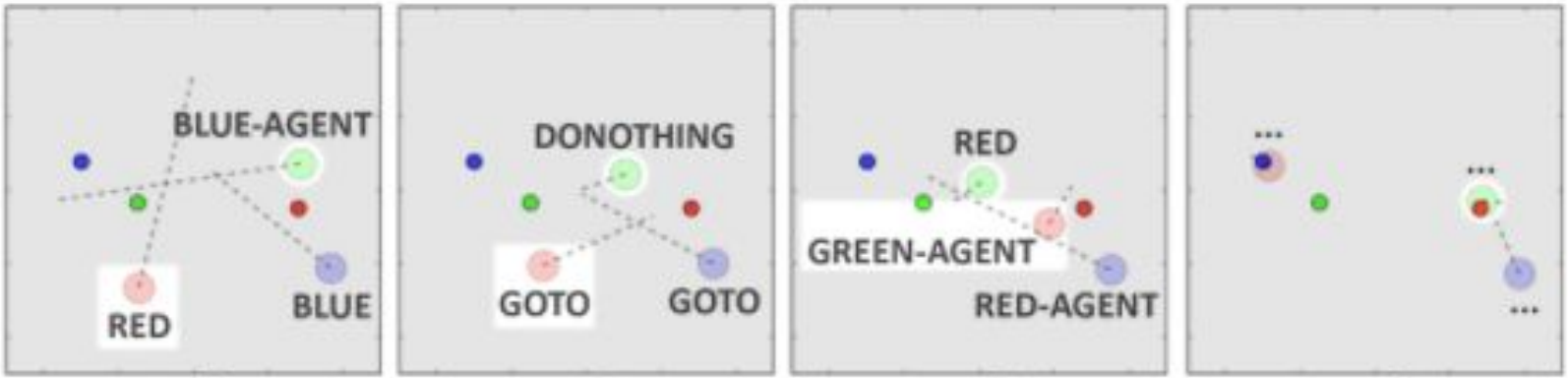
- Can be any number of agents and therefore streams
- Every agent and stream has its own processing module, with shared weights
- The outputs are pooled together with a softmax
- There is an auxiliary prediction reward for predicting goals of other agents



Experiment

- 3 types of actions
- Private goals
- No global position vectors
- Agents cannot see other agents
- Typical conversation

```
Green Agent: GOTO, GREEN, ...
Blue Agent: GOTO, BLUE,  ...
```



In the above step-by-step run, at t=0 the red agent says a word corresponding to the red landmark (center right), then at t=1 says a word that is equivalent to 'Goto', then in t=2 says 'green-agent'. The green-agent hears its instructions and immediately moves to the red landmark.

Non-verbal mode

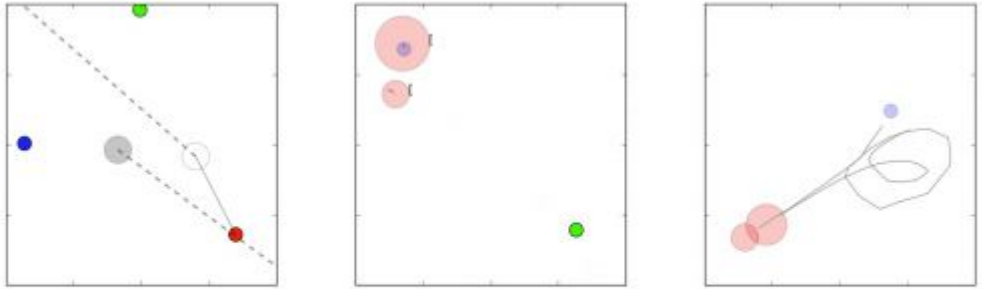


Figure 7: Examples of non-verbal communication strategies, such as pointing, guiding, and pushing.

Continued meet..

6th Feb, Saturday

Raghav

EMERGENT COMMUNICATION THROUGH NEGOTIATION.

Introduction

The aim of this paper is to show that interaction between different agents is necessary for emergence of novel protocols of communication.

OVERVIEW-

- A negotiation game is being played between two agents.
- They are in a semi-cooperative environment in which they have to negotiate with each other.
- They can communicate using two protocols -
  - First - Grounded in semantics of the game.
  - Second - Novel and a form of cheap talk.
- The agents can be selfish or prosocial based on the rewards they get.
- It is seen that selfish agents can't use the cheap talk to communicate. They perform well only with the grounded protocols.
- Prosocial agents are able to develop their own form of communication from the cheap talk protocol.

Negotiation Environment

Agents are presented with three types of items: peppers,cherries and strawberries.

At each round they are presented with-

1. A vector giving quantity of each item available for negotiation. $i \in \{0...5\}^3$
2. A utility vector for each agent specifying how rewarding each unit of the three item is for the agent. $u_j \in \{0...10\}^3$ This is private for both the agents.
3. At each time step agents can exchange three msgs
   a. A message using the cheap talk protocol. (that include random strings)
   b. A proposal message based on the semantics of the game, stating quantity of each item required by the agent. $p_t^j \in \{0...5\}^3$
4. Each agent acts at alternate time steps. With the agent A always starting first.
5. Each agent can terminate at time t with a special action accepting the proposal presented by the agent in the previous timestep.

Agent Sociality and Rewards

At the termination of the round by an agent at timestep t, both agent receive a reward given by -

$R_A = u_A \cdot (p^A_{t-1})$ $R_B = u_B \cdot (i - p^A_{t-1}).$

(Supposing that the episode was terminated by agent A)

If they are not able to reach to an agreement before the end of the round both are given no rewards.

A new combined reward is introduced to make agents prosocial. $R = R_A + R_B.$

The selfish agents get only their own reward whereas the prosocial agents are given the combined reward in addition.

Agent Architecture And Learning

At each timestep t, the proposer receives three inputs:

- The item context $c^j = [i; u_j]$, a concatenation of the item pool and the proposer's utilities.
- The utterance m_{t-1} produced by the other agent in the previous timestep $t-1$. If the linguistic channel is closed, this is simply a dummy message.
- The proposal p_{t-1} made by the other agent in the previous timestep $t-1$. If the proposal channel is closed, this is again a dummy proposal.

Then all three of them are converted to dense vectors using embedding tables.
We use two seperate tables -

1. For item context and the proposal
2. For the cheap talk utterance

Agent Architecture And Learning (contd.)

The three dense vectors obtained are encoded separately using an **LSTM** for each input.

This results in three vectors-

$$h_t^c,\ h_t^m \text{ and } h_t^p.$$

These are then concatenated and fed through a feedforward layer ending with ReLU activation, giving us h(t), the hidden state of agent at timestep t.

This hidden state is then used to determine the policies for the agent using three different networks.

Agent Architecture And Learning (contd.)

- π_{term} is the policy for the termination action. If this action is taken by an agent, both agents receive reward according to the last proposal made by the other agent. This is a binary decision, and we parametrise π_{term} as a single feedforward layer, with the hidden state as input, followed by a sigmoid function, to represent the probability of termination.
- π_{utt} is the policy for the linguistic utterances. This is parametrised by an LSTM, which takes the hidden state of the agent as the initial hidden state. For the first timestep, a dummy symbol is fed in as input; subsequently, the model prediction from the previous timestep is fed in as input at the next timestep, in order to predict the next symbol.
- π_{prop} is the policy for the proposals the agent generates. This is parametrised by 3 separate feedforward neural networks, one for each item type, which each take as input h_t and output a distribution over $\{0...5\}$ indicating the proposal for that item.

Agent Architecture And Learning (contd.)

The overall policy of the agent is a tuple of all the three policies mentioned previously.

The network is trained with the aim to find the optimal policy as -

$$\pi_i^* = \arg\max_{\pi_i} \mathop{\mathbb{E}}_{\tau \sim (\pi_A, \pi_B)} [R_i(\tau)] + \lambda H(\pi_i)$$

The parameters are updated using REINFORCE algorithms.

Topic...

Paper presentation by:

- Ayush
- Raghav
- Lokesh

Followed by discussion on:

1. Overall domain of communication in MARL
2. Intricacies and improvements upon the papers we have discussed
3. Potential ideas in each of the papers (which could be developed into a paper/workshop abstract)
4. Any revisit required, based on our discussions
5. Finally, deciding the next domain.

To facilitate this, before meeting on saturday, we need to:

1. Look(bird's eye view) at the papers by others and decide what we find interesting (recording could be helpful)
2. Select paper/papers which we find interesting from those and read them completely. I'm sure we will get some ideas of improvement, if not then we take up the "Future Directions" from the paper and discuss those.

Ideas:

Yash: Adding communication in SSDs