Winning an Election: On Emergent Strategic Communication in Multi-Agent Networks

Extended Abstract

Shubham Gupta
Indian Institute of Science
Bangalore, India
shubhamg@iisc.ac.in

Ambedkar Dukkipati
Indian Institute of Science
Bangalore, India
ambedkar@iisc.ac.in

ABSTRACT

Humans use language to collectively execute abstract strategies besides using it as a referential tool for identifying physical entities. In this paper, we study the role of emergent languages in discovering and implementing strategies. We formulate the problem using a voting game where two candidate agents contest in an election with the goal of convincing population members (other agents), that are connected to each other via an underlying network, to vote for them. To achieve this goal, agents are only allowed to exchange messages in the form of sequences of discrete symbols. Using our proposed framework, we answer the following questions: **(i)** Do the agents learn to communicate in a meaningful way? **(ii)** Does the system evolve as expected under various reward structures? **(iii)** How is the emergent language affected by the community structure in the network? To the best of our knowledge, we are the first to study emergence of communication among networked agents for discovering and implementing strategies.

CCS CONCEPTS

• **Computing methodologies** → **Multi-agent systems**; *Multi-agent reinforcement learning*;

KEYWORDS

emergent communication; multi-agent reinforcement learning

ACM Reference Format:
Shubham Gupta and Ambedkar Dukkipati. 2020. Winning an Election: On Emergent Strategic Communication in Multi-Agent Networks. In *Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), Auckland, New Zealand, May 9–13, 2020,* IFAAMAS, 3 pages.

1 INTRODUCTION

In the context of multi-agent reinforcement learning, several attempts at understanding the emergence of language have been made [3, 4, 6, 9, 10]. These approaches use variants of the Lewis signaling game [11] where agents develop a *grounded* language, i.e. words correspond to physical concepts, to maximize their rewards. But, humans also use language for collectively devising strategies [5], in which case, abstract concepts also play an important role. Recently, a few approaches that study the emergence of language for planning have also been proposed [1, 2, 13]. In this paper, we consider the second setting.

Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), B. An, N. Yorke-Smith, A. El Fallah Seghrouchni, G. Sukthankar (eds.), May 9–13, 2020, Auckland, New Zealand.

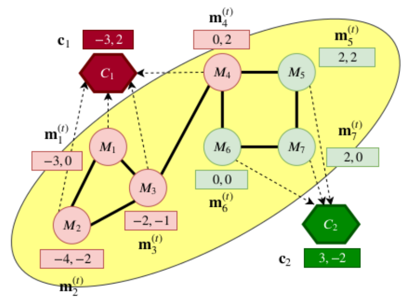

Figure 1: Game demonstration: Members (circles), candidates (hexagons) and their private *preference/propaganda* vectors have been shown. Each member M_i has been colored based on $\mathbf{F}_i^{(t)}$. The ellipse marks the network boundary. Members broadcast messages to their neighbors, candidates broadcast messages to their followers (dashed arrows).

We propose a voting game that involves n agents (which we call members) and two special agents (which we call candidates). Members are connected to each other via an underlying network. At each time step, members broadcast a message in the form of a sequence of discrete symbols to their immediate neighbors and similarly candidates broadcast a message to the members that follow them. After T time steps, voting is conducted where each member votes for exactly one candidate. We consider different objectives for agents in this game (e.g., maximize the vote count). One can say that, over time, the candidates have to *persuade* members to vote for them using the messages that they broadcast. However, as members also communicate with each other, other interesting strategies may also emerge (see Section 3).

The agents in our proposed game can either be competitive or cooperative. To the best of our knowledge, we are the first to study a setting where emergent communication is restricted along an underlying social network. We show that this leads to interesting insights on emergent strategies, languages and connections between language and network community structure.

2 VOTING GAME WITH COMMUNICATION

There are two types of agents in the game: n population members $\{M_1, M_2, \ldots, M_n\}$ connected to each other via an underlying social network and two candidates C_1 and C_2. The candidates contest an election, seeking votes from the members. The game consists of T *propaganda steps* followed by a *voting step*. Each candidate C_j has a fixed *propaganda vector* $\mathbf{c}_j \in \mathbb{R}^d$ and each member M_i has a time dependent *preference vector* $\mathbf{m}_i^{(t)} \in \mathbb{R}^d$. During each propaganda step t, each member M_i *follows* one of the two candidates. Let $\mathbf{F}_i^{(t)} \in \{1, 2\}$ denote the candidate being followed by member M_i at time t and $\mathbf{F}^{(t)} \in \{1, 2\}^n$ be a random vector whose i^{th} entry is $\mathbf{F}_i^{(t)}$. Members observe their own preference vector $\mathbf{m}_i^{(t)}$ and candidates observe the network adjacency matrix $\mathbf{A}$ and random vector $\mathbf{F}^{(t)}$ in addition to observing their own propaganda vector $\mathbf{c}_j$.

Let $\mathcal{V}$ be a set of n_{vocab} elements (vocabulary). A *communication action* selects a sequence of discrete symbols (a message) from the set $\mathcal{V}^{L_{\max}}$. Both candidates and members choose a message at each time t based on their observations. The message chosen by a member is broadcasted to its neighbors in the social network and the message chosen by a candidate C_j is broadcasted to all members M_i for which $\mathbf{F}_i^{(t)} = j$. These messages become part of the observation made by the receiving agent at time $t + 1$ and are used for taking actions. Each member additionally takes a *modification action* at each time step which selects a vector $\hat{\mathbf{m}}_i^{(t)}$ and scalar $\lambda_i^{(t)}$ that are used for modifying the preference vector $\mathbf{m}_i^{(t)}$ of the member as follows: $\mathbf{m}_i^{(t+1)} = (1 - \lambda_i^{(t)})\mathbf{m}_i^{(t)} + \lambda_i^{(t)}\hat{\mathbf{m}}_i^{(t)}$. Here, $\lambda_i^{(t)} \in (0, \epsilon)$ and $\epsilon \in (0, 1)$ is a hyperparameter. The environment randomly samples a value of $\mathbf{F}_i^{(t)}$ by passing the vector $\left(||\mathbf{m}_i^{(t)} - \mathbf{c}_1||_2^2/d, ||\mathbf{m}_i^{(t)} - \mathbf{c}_2||_2^2/d\right)$ through Gumbel-Softmax [7, 12] to get a one-hot encoded vector which represents the choice made by member M_i.

After T propaganda steps, voting is conducted. Each member M_i votes for exactly one candidate C_1 or C_2. Let $\mathbf{V}_i \in \{1, 2\}$ denote the vote cast by member M_i. As in the case of $\mathbf{F}_i^{(t)}$, the value of $\mathbf{V}_i$ is sampled by passing the vector $\left(||\mathbf{m}_i^{(T+1)} - \mathbf{c}_1||_2^2/d, ||\mathbf{m}_i^{(T+1)} - \mathbf{c}_2||_2^2/d\right)$ through Gumbel-Softmax to get a one-hot encoded vector. The reward functions r_i, for all agents i, determine whether the agents will be cooperative or competitive. Candidates follow separate policies but all members share the same policy. However, since the policy used by members is a function of their preference vectors, the members can take different actions and hence the setup is fairly expressive. All members and candidates share the same vocabulary, message encoder and decoder (which we collectively call the *communication engine*) for communicating with each other. We parameterize all of these components using neural networks. The setup is end-to-end differentiable due to the use of Gubmel-Softmax and hence backpropagation algorithm can be directly used. A toy example has been presented in Figure 1.

3 EXPERIMENTAL RESULTS

We experimented with two rewards for candidates. Let N_j be the number of votes obtained by C_j. In the first case, the reward for candidate C_j is given by N_j. This makes the candidates competitive. In the second case, candidates are cooperative: for C_1 the reward is N_1 but for C_2 the reward is $-N_2$. The reward for member M_i is computed as: $r_i = -||\mathbf{m}_i^{(T+1)} - c(i)||_2^2$, where $c(i) = \mathbf{c}_j$ for j such that $\mathbf{V}_i = j$. The reward given to member policy network is the average of rewards obtained by members. We used the largest connected component of the Network Science Collaborations network [14] as the underlying social network. It has 379 nodes and 914 edges.

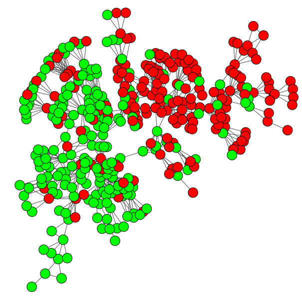

Figure 2: Clustering of nodes in graph based on their language usage. Colors represent different clusters. It can be seen that communities based on language usage overlap with structural communities.

We also experimented with randomly sampled networks using the random geometric graph model [15].

To demonstrate that the agents are learning something meaningful we observe the effect of placing each trained candidate in an environment where only that candidate is active, i.e. the other candidate is not allowed to broadcast messages. Note that all members can still exchange messages irrespective of the candidate they follow. We observed that the active candiadate wins the election when competitive rewards are used and C_1 wins the election when cooperative rewards are used irrespective of the active candidate. Thus, the agents have learned the expected behavior.

We also performed analysis of the language generated by candidates and members. We created a $n \times n_{\text{vocab}}$ dimensional member-symbol matrix which we denote by $\mathbf{W}$. $\mathbf{W}_{ij}$ counts the number of times M_i uttered the j^{th} symbol. We converted this matrix to a tf-idf matrix [8] and then clustered its rows using spectral clustering (using cosine similarity). Figure 2 shows the result of clustering the members from the Network Science Collaborations network based on the rows of matrix $\mathbf{W}$. Although these clusters were discovered based on language usage, they naturally correspond to underlying structural communities. This implies that members that are connected to each other develop a language of their own which may be different from language developed in other communities.

4 CONCLUSION

We studied a voting game where the agents learn to communicate with each other to develop intelligent strategies that maximize their rewards. Further, this communication is only allowed over an underlying network that connects the agents. Due to the flexibility of our proposed framework, many interesting questions can be studied under it and we believe that our work will serve as a stepping stone for future research in this direction. For example, members may compete amongst themselves to secure the highest number of votes as opposed to having designated special candidate agents. Could such a setup explain why communities form in real world networks? Are they a result of globally competing agents with a local neighborhood based reward structure? What if agents are given the ability to privately communicate with each other without broadcasting a message?

REFERENCES

[1] Ben Bogin, Mor Geva, and Jonathan Berant. 2018. Emergence of Communication in an Interactive World with Consistent Speakers. *Emergence Communication Workshop at NeurIPS* (2018).

[2] Kris Cao, Angeliki Lazaridou, Marc Lanctot, Joel Z Leibo, Karl Tuyls, and Stephen Clark. 2018. Emergent Communication through Negotiation. In *International Conference on Learning Representations*. https://openreview.net/forum?id=Hk6WhagRW

[3] Abhishek Das, Satwik Kottur, Jose M. F. Moura, Stephan Lee, and Dhruv Batra. 2017. Learning Cooperative Visual Dialog Agents with Deep Reinforcement Learning. In *The IEEE International Conference on Computer Vision (ICCV)*. 2951–2960.

[4] Jakob Foerster, Ioannis Alexandros Assael, Nando de Freitas, and Shimon Whiteson. 2016. Learning to Communicate with Deep Multi-Agent Reinforcement Learning. In *Advances in Neural Information Processing Systems*. Vol. 29. Curran Associates, Inc., New York, 2137–2145.

[5] Yuval Noah Harari. 2015. *Sapiens: A Brief History of Humankind*. Harper, New York.

[6] Serhii Havrylov and Ivan Titov. 2017. Emergence of Language with Multi-agent Games: Learning to Communicate with Sequences of Symbols. In *Advances in Neural Information Processing Systems*. Vol. 30. Curran Associates, Inc., New York, 2149–2159.

[7] Eric Jang, Shixiang Gu, and Ben Poole. 2017. Categorical Reparameterization with Gumbel-Softmax. In *International Conference on Learning Representations*. https://openreview.net/forum?id=rkE3y85ee

[8] Dan Jurafsky and James H. Martin. 2009. *Speech and Language Processing : An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition - Chapter 23*. Pearson Prentice Hall.

[9] Angeliki Lazaridou, Karl Moritz Hermann, Karl Tuyls, and Stephen Clark. 2018. Emergence of Linguistic Communication from Referential Games with Symbolic and Pixel Input. In *International Conference on Learning Representations*. https://openreview.net/forum?id=HJGv1Z-AW

[10] Angeliki Lazaridou, Alexander Peysakhovich, and Marco Baroni. 2017. Multi-Agent Cooperation and the Emergence of (Natural) Language. In *International Conference on Learning Representations*. https://openreview.net/forum?id=Hk8N3Sclg

[11] David Lewis. 1969. *Convention: A Philosophical Study*. Harvard University Press, Cambridge, MA.

[12] Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. 2017. The Concrete Distribution: A Continuous Relaxation of Discrete Random Variables. *International Conference on Learning Representations*.

[13] Igor Mordatch and Pieter Abbeel. 2018. Emergence of Grounded Compositional Language in Multi-Agent Populations. In *Thirty-Second AAAI Conference on Artificial Intelligence*. 1495–1502. https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17007

[14] M. E. J. Newman. 2006. Finding Community Structure in Networks using the Eigenvectors of Matrices. *Phys. Rev. E* 74, 036104 (2006).

[15] Mathew Penrose. 2003. *Random Geometric Graphs*. Oxford University Press, Oxford.