The spontaneous emergence of discrete and compositional messages

Anonymous ACL submission

Abstract

blah blah blah

1 Introduction

In a signalling game, artificial agents communicate to achieve a common goal: a sender sees some piece of information and produces a message, this message is then sent to a receiver that must take some action (Lewis, 1969; Skyrms, 2010). If the action is appropriate, the whole communication stream, and in particular the choice of the message, is reinforced. For instance, in a referential game, sender and receiver see a set of objects, and the sender must send a message to the receiver, so that the receiver can pick up the right object, as determined in advance for the sender, but unbeknownst to the receiver (Lazaridou et al., 2017, 2018; Havrylov and Titov, 2017; Chaabouni et al., 2019).

This setting has been used to study the factors influencing the emergence of various fundamental properties of natural language, such as compositionality (Kirby et al., 2015; Franke, 2016; Steinert-Threlkeld, 2016; Mordatch and Abbeel, 2018; Lazaridou et al., 2018; Choi et al., 2018).

2 Function Games

We here introduce a general communication game setting, which we call Function Games. Our games contain three basic components: (i) a set of contexts C, (ii) a set of actions A, (iii) a family of functions F, from contexts to actions. One play of a Function Game game runs as follows:

1. Nature chooses $f \in F$ and a context $c \in C$.
2. Sender sees the context c and f. I like f(c) here, but f is a bit more appropriate. What do you all think? Nur: Given the situation, f is the only choice, no?
3. Sender sends a message m to Receiver.
4. Receiver sees *a possibly different* context c' and the message m and chooses an action a'.
5. Both are 'rewarded' iff $a' = f(c')$.

Two concrete interpretations will be helpful in illustrating the various components.

Generalized referential games. A reference game is one in which Sender tries to get Receiver to pick the correct object out of a given set (Skyrms, 2010; Lazaridou et al., 2017, 2018; Havrylov and Titov, 2017; Chaabouni et al., 2019). Here, contexts are sets of objects (i.e. an $m \times n$ matrix, with m objects represented by n features). Normally (though we will drop this assumption later), $c' = \texttt{shuffled}(c)$: Sender and Receiver see the same objects, but in a different arrangement. Actions are the objects, and the functions $f \in F$ are *choice functions*: $f(c) \in c$ for every context c.

Belief update games. Contexts can represent possible belief states for the agents. Letting $A = C$, the functions will then be 'belief update' functions, representing e.g. how to update an agent's beliefs in the light of learning a new piece of information. What should we cite here? Something from dynamic semantics?

3 Experiment

Because we are interested in the simultaneous emergence both of discrete signals and of compositional messages, we use a Function Game called the Extremity Game designed to incentivize compositionality (Steinert-Threlkeld, 2019). This is a generalized referential game, where objects are represented as n-dimensional vectors, with each value corresponding to the degree to which it has a gradable property. For instance, objects could be shaded circles, with two values, one for their diameter and one for their darkness. For the functions, we set $F = \{\arg\min_i, \arg\max_i : 0 \leq i < n\}$. These may incentivize the emergence of communication protocols with messages like 'small + EST' and 'dark + EST'.

3.1 Model

Our model resembles an encoder-decoder architecture, with the Sender encoding the context/target pair into a message, and the Receiver decoding the message (together with its context c') into an action. Both the encoder and decoder are multi-layer perceptrons with two hidden layers of size 64 and rectified linear activation (ReLU) (Nair and Hinton, 2010; Glorot et al., 2011). A smaller, intermediate layer without an activation function bridges the encoder and decoder and represents the transformation of the input information to messages. Figure 1 depicts this architecture.

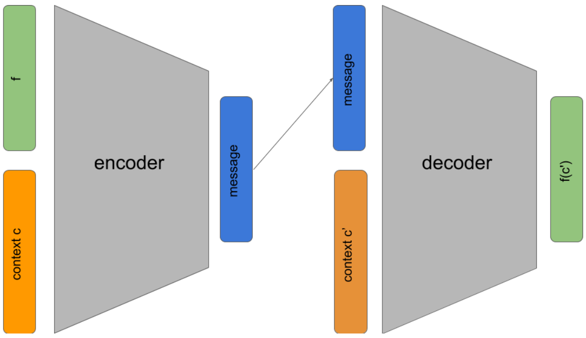

Figure 1: Model architecture caption Do we really need this? It isn't super informative and might use too much space. I can also do something in TikZ if we think it's important and that this one is ugly.

3.2 Game Parameters

In our experiments, we manipulated the following parameters of the Extremity Game:

- Context strictness. In *strict* contexts, every object is the $\arg\max$ or $\arg\max$ of exactly one dimension. This means that there is a one-to-one (and onto) correspondence between F and $A = C$.[1] In *non-strict* contexts, no such restriction is imposed.

  We considered strict contexts with 10 objects (i.e. each object consists of 5 dimensions) and non-strict contexts with 5, 10, and 15 objects.

- Context identity. In the *shared* setting, Receiver sees a shuffled version of Sender's context ($c' = \mathtt{shuffled}(c)$). In the *non-shared* setting, Receiver's context c' is entirely distinct from Sender's. This may incentivize compositional messages, since Sender cannot rely on the raw properties of the target object in communication.

- Object size: in all experiments, objects had 5 dimensions.

- Latent space (message) dimension: in all experiments, the latent space had 2 dimension[2].

3.3 Training Details

By using a continuous latent space, the entire model, including the communication channel, is differentiable and so can be trained end-to-end using backpropagation to compute gradients. We used the Adam optimizer (Kingma and Ba, 2015) with learning rate 0.001, $\beta_1 = 0.9$, and $\beta_2 = 0.999$. The model was trained for 5,000 epochs by feeding the network with mini-batches of 64 contexts concatenated with one-hot function selectors. The network's loss is taken as the MSE between the target object $f(c')$ and the object generated by the Receiver. For each setting of the above parameters, we ran 20 trials with different random seeds.

Code and data will be made available once the paper can be de-anonymized.

4 Results

We measure the communicative success of the network by calculating the accuracy of recovering the correct object from c'. The Receiver's prediction is considered correct if its output is closest to $f(c')$ than to all other objects in c'. The recovery accuracy of the different settings is reported in Table 2. While the network handles well the $c \neq c'$ setting (*unshared context*), the model struggles with non-strict contexts. Note that although accuracy is not 100%, it is still well above chance, since e.g. for a context of 10 objects random guessing will yield an expected accuracy of 10%. Table 1 shows the mean loss and recovery accuracy for a network before and after training. The randomly-initialized network's accuracy is at the expected chance level for a 10-object setting.

4.1 Discrete signals

The model's ability to discretizie the communication is measured by inspecting the information generated by the intermediate layer. Figure 2 depicts message vectors sampled from this layer. The

[1]These are the contexts used in (Steinert-Threlkeld, 2019).

[2]The model also performs well with messages of size 1, not reported here. Using messages of size 2 makes it easier to inspect the latent space using 2-D visualization.

Commented out the actual figures because I can't compile without them.

Figure 2: (a) Messages sampled from latent space of a trained network, for objects of size 5 and contexts of 10 objects. (b) Messages sampled from an untrained network. Colors represent the $f_i \in F$ input part of the Sender.

	Trained	**Untrained**
Accuracy	$63.78\% \pm 0.02$	$9.92\% \pm 0.01$
Loss (MSE)	0.04 ± 0.00	0.35 ± 0.03

Table 1: Mean object recovery accuracy and prediction loss before and after training, for objects of size 5 in a shared, strict context setting (10 objects per context).

	Shared	**Non-shared**
Strict context	$63.78\% \pm 1.63$	$60.22\% \pm 1.56$
Non-strict, 5 objects	$49.37\% \pm 1.67$	$43.55\% \pm 1.69$
Non-strict, 10 objects	$33.06\% \pm 1.47$	$31.89\% \pm 1.63$
Non-strict, 15 objects	$27.58\% \pm 1.30$	$27.95\% \pm 1.24$

Table 2: Object recovery accuracy for the different model settings.

	Shared	**Non-shared**
Strict context	1.00 ± 0.00	0.90 ± 0.09
Non-strict, 5 objects	0.99 ± 0.02	0.54 ± 0.15
Non-strict, 10 objects	1.00 ± 0.00	0.99 ± 0.01
Non-strict, 15 objects	1.00 ± 0.00	1.00 ± 0.00

Table 3: Message clusterization F1 scores.

	Shared	**Non-shared**
Strict context	$63.39\% \pm 1.45$	$55.37\% \pm 3.43$
Non-strict, 5 objects	$46.94\% \pm 1.70$	$29.40\% \pm 5.59$
Non-strict, 10 objects	$32.63\% \pm 1.43$	$31.51\% \pm 1.62$
Non-strict, 15 objects	$28.24\% \pm 1.11$	$27.94\% \pm 1.20$

Table 4: Object prediction accuracy using average message from each function cluster.

same is depicted for an untrained network with randomized weights, where the messages are not yet clustered.

We make the discretization measure concrete by calculating the F1 score between the cluster labels for transmitted messages and the target functions for which they were generated. For this, an unsupervized clustering algorithm is first applied to the message vectors, giving an expected number of clusters (DBSCAN, Ester et al., 1996, with $epsilon = 0.5$). The cluster labels are then matched with the respective function labels by taking the most common function in each cluster. If message clusters are well separated from one another, the labeling will have less to no confusion and an F1 score closer to 1. The F1 scores for the different model settings are given in Table 3. The model reached near-optimal clusterization measures for both shared and non-shared contexts, and for both strict and non-strict contexts.

Given the clusterization of the message space, we are able to sample unseen messages from each cluster, and test the Receiver's perception of 'artificial' messages. 10 messages are sampled from each cluster, and their average vector is fed to the Receiver. The output object accuracy for these unseen messages is given in Table 4. The model achieves recovery accuracy similar to when messages are generated using actual inputs. This can be paralleled with the phenomenon of Categorical Perception, which describes how continuous signals, such as phonemes in an acoustic space, are perceived as stable and discrete, even when the signal is gradually shifted in the continuous space.

4.2 Compositionality

5 Discussion

6 Conclusion

References

Rahma Chaabouni, Eugene Kharitonov, Emmanuel Dupoux, and Marco Baroni. 2019. Anti-efficient encoding in emergent communication. In *Proceedings of the 33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*.

Edward Choi, Angeliki Lazaridou, and Nando de Freitas. 2018. Compositional Obverter Communication Learning from Raw Visual Input. In *Inter-*

national Conference of Learning Representations (ICLR 2018)*, pages 1–18.

Martin Ester, Hans-Peter Kriegel, Jrg Sander, and Xiaowei Xu. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231.

Michael Franke. 2016. The Evolution of Compositionality in Signaling Games. *Journal of Logic, Language and Information*.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep Sparse Rectifier Neural Networks. In *14th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 315–323.

Serhii Havrylov and Ivan Titov. 2017. Emergence of Language with Multi-agent Games: Learning to Communicate with Sequences of Symbols. In *Proceedings of the 31st Conference on Neural Information Processing Systems (NeurIPS 2017)*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *International Conference of Learning Representations (ICLR)*.

Simon Kirby, Monica Tamariz, Hannah Cornish, and Kenny Smith. 2015. Compression and communication in the cultural evolution of linguistic structure. *Cognition*, 141:87–102.

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 of Learning Representations (ICLR 2018)*.

Angeliki Lazaridou, Alexander Peysakhovich, and Marco Baroni. 2017. Multi-Agent Cooperation and the Emergence of (Natural) Language. In *International Conference of Learning Representations (ICLR2017)*.

David Lewis. 1969. *Convention*. Blackwell.

Igor Mordatch and Pieter Abbeel. 2018. Emergence of Grounded Compositional Language in Multi-Agent Populations. In *The Thirty-Second AAAI Conference on Artificial Intelligence (AAAI 2018)*.

Vinod Nair and Geoffrey E Hinton. 2010. Rectified Linear Units Improve Restricted Boltzmann Machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*.

Brian Skyrms. 2010. *Signals: Evolution, Learning, and Information*. Oxford University Press.

Shane Steinert-Threlkeld. 2016. Compositional Signaling in a Complex World. *Journal of Logic, Language and Information*, 25(3):379–397.

Shane Steinert-Threlkeld. 2019. Paying Attention to Function Words. In *Emergent Communication Workshop @ NeurIPS 2018*.