Understanding Redundancy in Discrete Multi-Agent Communication

Anonymous Author(s)
Affiliation
Address
email

Abstract

Through providing agents with the capacity to learn sample-efficient and generalisable communications protocols, we may enable them to more effectively cooperate in real-world tasks. In this paper, we consider this in the context of discrete decentralised multi-agent reinforcement learning to provide insights into the impact of the often overlooked size of the message set. Within a referential game, we find that over-provisioning the message set size leads to improved sample efficiency, but that these policies tend to maintain a high-degree of redundancy, often utilising multiple messages to refer to each label in the dataset. We hypothesise that the additional redundancy within these converged policies may have implications for generalisation and experiment with methodologies to gradually reduce redundancy while maintaining sample-efficiency. To this end, we propose a linearly-scheduled entropy regulariser which encourages an agent to initially maximise the utilisation of the available messages but, as training progresses, it tries to minimise it. Through this mechanism, we achieve a comparable sample efficiency whilst converging to a model with significantly reduced redundancy and that generalises more effectively to previously unseen data.

1 Introduction

Intuitively, in most cooperative tasks agents can benefit from communicating. The ability to interact allows them to share information with their collaborators, which can facilitate better cooperative decisions. Through studying the emergent properties of this type of communication, we hope to equip cooperative agents with the necessary tools to quickly develop protocols which generalise to unseen states.

Within this work, we focus on the emergence of discrete communication protocols in Multi-agent Reinforcement Learning (MARL) with a particular interest on Independent Learners (IL) [1]. This challenging domain has previously been considered within a number of previous works [2, 3, 4]. However, little attention has been paid to the impact of the dimensionality of the message set on the messaging protocol, which could lead to unexpected behaviours. As an example, we explore this within an MNIST-based referential game (shown in Figure 3). The game requires two agents, a speaker and a listener, to derive a protocol to facilitate communication of the speakers observed digit to the listener via a set of discrete messages, M. We experiment with messages set sizes of 10, 20, 30 and 40. The utilisation of MNIST [5] should mean that 10 messages are sufficient, but we find that over-allocation improves sample-efficiency on training data. Interestingly the derived policies tend to retain a high-degree of redundancy and typically utilise multiple messages to refer to the same digit.

We further build upon this result and investigate methods to reduce the redundancy within the protocol. By doing so, we hope to build more robust representations which better capture distinct concepts that a speaker (or agent more generally) may wish to communicate. We achieve this through introduction of

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

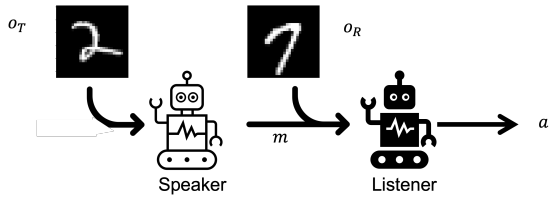


Figure 1: MNIST-based Referential Game comprised of two agents, a speaker and a listener.

a linearly-scheduled entropy regulariser into the speaker's loss function. At the beginning of training, this term acts to maximise the utilisation of the message set which has previously been demonstrated to improve performance in this setting [4]. As training progresses, we linearly reduce this incentive. Eventually, we switch from entropy maximisation to minimisation in order to encourage our speaker to reduce its utilisation of the message set, achieving thereby our desired reduction in effective cardinality. We empirically validate this approach and observe an improvement in generalisation.

This paper is structured as follows. In Section 2 we describe the existing works in this area. We then introduce our methodology in Section 3. Our experimentation setup and results are introduced in Section 4. We then discuss our results, their implications and outline a direction for future work in Section 3.3.

2 Related Work

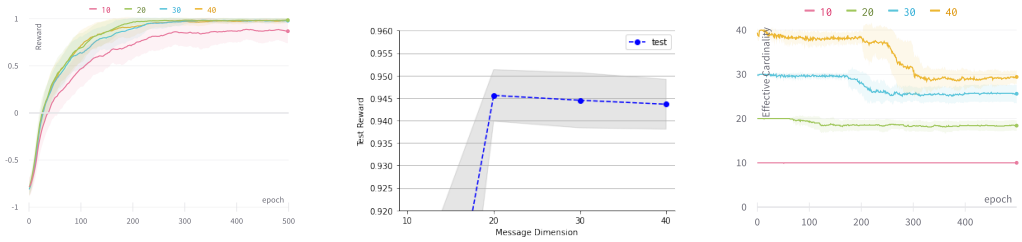
Enabling communication among agent's has attracted significant attention within the multi-agent reinforcement learning (MARL) literature [6]. A number of methods have been proposed including those that utilise GNNs [7, 8] and other forms of differentiable communication channels [2]. Within this work, we focus on discrete communication channels and independent learners (IL) [1] which has been previously considered by [3, 4]. The use of IL prohibits differentiable communication channels and parameter sharing among agents which makes learning more difficult but is considered to be a more biologically plausible method for learning. It also easily permits extension to situations where a communicating agent may not be differentiable. Our work differs from existing work as it explores the impact of message dimensionality within discrete communication in MARL and experiments with methodologies to reduce excessive redundancy.

3 Method

In Section 3.1 we define our setting. As the aim of this work is to investigate the impact of message set size in MARL, in Section 3.2, we demonstrating that over-provisioning M results in improved sample-efficiency, but that the policies tend to maintain a higher than necessary level of redundancy in their messaging protocols. We then propose a methodology to address this problem in Section 3.3.

3.1 Setting

We apply the MARL approach defined in this paper to an N-player partially observable stochastic game [9]. This is defined by the tuple $G = \{\mathcal{I}, \mathcal{S}, \{A^i\}, \{M^i\}, \{O^i\}, \mathcal{P}, \Omega, \{R_i\}\}$. Where, $\mathcal{I}$ is a finite set of agents indexed by $\{i, \ldots, N\}$, $\mathcal{S}$ is the state space, A^i and M^i represent the finite set of actions and messages that are available to agent i. We refer to the joint action and message space as $\mathcal{A} = A^1 \times \ldots \times A^n$ and $\mathcal{M} = M^1 \times \ldots \times M^n$. At each time-step agent i, receives a partial observation $o^i \in O^i$ which is defined by the observation function $\Omega : \mathcal{S} \times \mathcal{I} \rightarrow O$. Additionally, all agents also receive all messages sent by other agents in the previous time-step which we refer to as $m^{-i} \in \mathcal{M}^{-i}$. Agent's use this information to select an action $a^i \in A^i$ according to $\pi_{i,a}$ and a discrete message $m \in M_i$ according to the policy, $\pi_{i,m}$. They then experience a state transition $T : S, \mathcal{A} \rightarrow S$ and receive a reward $R^i : S, \mathcal{A} \rightarrow \mathcal{R}$. Agents are tasked with finding action policies $\pi_{i,a} : (o, m) \rightarrow A_i$ and a message policy $\pi_{i,m} : (o, m) \rightarrow M_i$ which maximise the cumulative discounted reward they receive.



(a) Performance on training data (b) Performance on test data (c) Effective Cardinality of speaker

Figure 2: Figures showing the impact of multiple message set cardinalities. Plotted results show mean and 95% confidence interval.

3.2 Problem Demonstration

We demonstrate the problem in an MNIST-based referential game shown in Figure 1. A referential game consists of two agents a speaker, s, and listener, l. Both agents observe O^s and O^l which are sampled from the dataset. The agents' undertake a cooperative task that requires the listener to sum the digits which the observations represent. If the correct answer is given both agents get a reward of 1, otherwise they get a reward of -1. The only mechanism to achieve this is through effective utilisation of the uni-directional communications channel which connects s to l. The channel permits s to transmit a single-message $m^s \in M^s$ to l. Successful completion of this task requires s and l to agree a mapping from $O^s \rightarrow M$ and for the listener to be able to identify O^l. We instantiate the speaker, $\pi_{s,\theta}$, and listener, $\pi_{l,\phi}$ as deep neural networks. Both agents are trained individually independently utilising REINFORCE [10]. We additionally utilise speaker and listener biases described by [4] to promote the emergence of communication. We conduct experiments with 10, 20, 30 and 40 messages each repeated with 10 random seeds. We use the same hyperparamaters defined within [4] and implement our code within PyTorch Lightning [11][1].

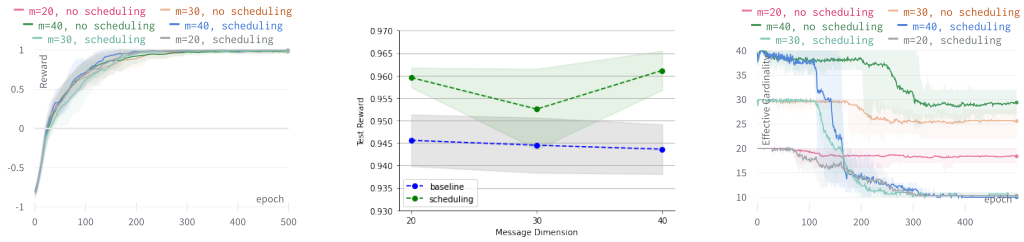
The performance of different cardinalities of M are shown in Figure 2. It is apparent from Figure 2a that over-provisioning of M results in improved sample efficiency when compared to $|M| = 10$. Over-allocation of M may afford a degree of flexibility within learning and reduce the difficulty of the task. The extra capacity could allow the speaker to represent more distinct members of an image class as separate messages, whereas $M = 10$ requires the speaker to learn that these images represent the same concept. This is supported by Figure 2c, which shows the *effective cardinality* of the message protocol. We define *effective cardinality* in eq. (1) which tells us which messages are being used by the speaker on a large batch of validation data, B. It illustrates how speakers tend to converge to messaging protocols which utilise a subset of M which is notably larger than the minimum viable size of 10. We expected that this may manifest in a reduction in test performance – this is supported by Figure 2b which shows a negative correlation between test performance and the remaining cardinality.

$$c(B) = \sum_{m \in M} \mathbb{1}[iff m \in B] \tag{1}$$

3.3 Scheduling Entropy Regularisation

By addressing the excessive redundancy in the speaker's converged messaging protocols, we hope to improve generalisation. We consider the speaker-side bias proposed within [4] as our starting point and further build upon them. In their standard form, these act to incentivise a speaker to maximise the mutual information between messages and observations. Practically, this is implemented into the speaker's loss through Equation (2). Where $\overline{\pi^i_M}$ is the average of the message policy over a batch of experience, and, λ and $\mathcal{H}_{target}$ are both hyperparameters. This encourages the speaker to output messages with uniform probability on average, but are not random with respect to an observation.

[1] Code will be made for publication



(a) Performance on training data (b) Performance on test data (c) Effective Cardinality of speaker

Figure 3: Figures showing the performances of scheduling method. Plotted results show mean and 95% confidence interval.

$$L_{ps}(\pi^i_M, o^i) = -\mathbb{E}(\lambda\mathcal{H}(\overline{\pi^i_M}) - (\mathcal{H}(m^i_t|o_t) - \mathcal{H}_{target})^2) \tag{2}$$

We speculate that maximisation of $\mathcal{H}(\overline{\pi^i_M})$ encourages the additional redundancy as it may provide a mechanism to encourages the speaker to represent more distinct members of an image class as separate messages. As training progresses and the speaker's competence begins to increase, we believe that it may be desirable to encourage it to minimise entropy and hopefully reducing the utilisation of the message set. We achieve this by modifying the behaviour of λ and making it a function of the speaker's experience as shown in Equation (3). Where, t is specified in epochs and c is a hyperparameter which controls the point at which we switch from entropy maximisation to minimisation.

$$\lambda(t) = \lambda \times max\left(\left[1 - \left(\frac{t}{c}\right), -1\right]\right) \tag{3}$$

4 Experiments and Results

We repeat the same experimentation in Section 3. The implementation remains the same, with minor modification to facilitate the scheduling given in Equation (3). We run 3 random seeds for 500 epochs and conduct a grid-search across $|M| = [20, 30, 40]$ and $c = [200, 300, 400]$. Our results are shown in Figure 3, where we include all values of $|M|$ for our best performing value of c which is 300.

Figure 3a demonstrates that the sample efficiency would appear to be mostly equivalent. Closer inspection does reveal that there is a slight improvement in converged performance on training data from the scheduling regulariser. Figure 3b shows that the introduction of the $\lambda(t)$ generally improves generalisation on unseen test data. Within Figure 3c, we show the effective cardinality which demonstrates that we have achieved the desired impact where it typically converges to a value close to 10 which is consistent with the MNIST dataset.

5 Discussion and Future Work

Our results support our hypothesis that addressing excessive redundancy can improve generalisation performance as shown in Figure 3b. The scheduled entropy regularisation, $\lambda(t)$, facilitates this and can be shown to significantly reduce the excess redundancy in the protocol as shown in Figure 3c. We believe that this method is effective as it enables for modification of the difficulty of the communication learning task throughout training through manipulation of the speaker's utilisation of its model's capacity. This view would may elude to parallels that exist with curriculum learning methods [12]

We believe that this method represents an interesting step towards generalisation within communicating MARL. Our current investigations were restricted to a referential game due to it being relatively amenable to analysis. Most usefully, it allowed us to know the minimal number of messages required for the task and for the agents' to concentrate on communication without distraction by other environmental tasks. Our future work intends to extend our experimentation to more complex temporally-extended environments where communications protocols may be more abstract.

References

[1] Ming Tan. "Multi-agent reinforcement learning: Independent vs. cooperative agents". In: *Proceedings of the tenth international conference on machine learning*. 1993, pp. 330–337.

[2] Jakob N. Foerster et al. "Learning to Communicate with Deep Multi-Agent Reinforcement Learning". In: *Proceedings of the 30th International Conference on Neural Information Processing Systems*. NIPS'16. Barcelona, Spain: Curran Associates Inc., 2016, pp. 2145–2153. ISBN: 9781510838819.

[3] Natasha Jaques et al. "Social influence as intrinsic motivation for multi-agent deep reinforcement learning". In: *International Conference on Machine Learning*. PMLR. 2019, pp. 3040–3049.

[4] Tom Eccles et al. "Biases for Emergent Communication in Multi-agent Reinforcement Learning". In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*. 2019, pp. 13111–13121.

[5] Yann LeCun and Corinna Cortes. "MNIST handwritten digit database". In: (2010). URL: `http://yann.lecun.com/exdb/mnist/`.

[6] Angeliki Lazaridou and Marco Baroni. "Emergent multi-agent communication in the deep learning era". In: *arXiv preprint arXiv:2006.02419* (2020).

[7] Jiechuan Jiang et al. "Graph Convolutional Reinforcement Learning". In: *ICLR*. 2020.

[8] Abhishek Das et al. "Tarmac: Targeted multi-agent communication". In: *International Conference on Machine Learning*. PMLR. 2019, pp. 1538–1546.

[9] Eric A Hansen, Daniel S Bernstein, and Shlomo Zilberstein. "Dynamic programming for partially observable stochastic games". In: *AAAI*. Vol. 4. 2004, pp. 709–715.

[10] Ronald J Williams. "Simple statistical gradient-following algorithms for connectionist reinforcement learning". In: *Machine learning* 8.3 (1992), pp. 229–256.

[11] William Falcon et al. "PyTorch Lightning". In: *GitHub. Note: https://github.com/PyTorchLightning/pytorch-lightning* 3 (2019).

[12] Sanmit Narvekar et al. "Curriculum Learning for Reinforcement Learning Domains: A Framework and Survey". In: *Journal of Machine Learning Research* 21.181 (2020), pp. 1–50. URL: `http://jmlr.org/papers/v21/20-212.html`.