Ease-of-teaching and Language Structure in Emergent Communication

by

Fushan Li

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computing Science

University of Alberta

© Fushan Li, 2019

Abstract

Artificial agents have been shown to learn to communicate when needed to complete a cooperative task. Some level of language structure (e.g., compositionality) has been found in the learned communication protocols. This observed structure is often the result of specific environmental pressures during training. By introducing new agents periodically to replace old ones, sequentially and within a population, we explore a new pressure — ease of teaching — and show its impact on the structure of the resulting language.

While in theory randomness is an intrinsic property, in practice, randomness is incomplete information.

– Nassim Nicholas Taleb, The Black Swan, 2007.

Acknowledgements

The most important person I cannot appreciate more is my supervisor, Michael Bowling. He offers me opportunities to grow in many aspects and guides me all along the way of being a better researcher. He has also influenced me to think positively and act more productively. I feel really lucky to have Mike as my supervisor and appreciate his time and patience to help me grow. The next person I want to thank is Richard Sutton. He opens the door for my research interest in reinforcement learning. I remember many words from Rich, which are inspiring for me to rethink about human intelligence. I also want to thank my other supervisor, Dale Schuurmans, who provides me wise insights at my early research stage. I want to thank the useful discussion on this thesis with Angeliki Lazaridou, Jakob Foerster and Fei Fang. I thank Yash Satsangi for proofreading and giving comments on my writing. Special thanks to people, who have provided me generous help and encouragement in my academic life, including Marlos Machado, Zaheen Ahmad, Zach Holland, Jesse Farebrother, Chen Ma, Kris De Asis, Levi Lelis, et al. Without naming a long list, I appreciate the friendships I have made at the University of Alberta and the friends I still talk to occasionally even though they are not in Edmonton. They give me joy now and then and make me feel not alone. Last but not least, I would like to thank the company and the support from my boyfriend Junfeng Wen as well as from my family during my Masters.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Communication among agents is often necessary in partially observable multi-agent cooperative tasks. Recently, communication protocols have been automatically learned in pure communication scenarios such as referential games [12, 19, 35] as well as alongside behaviour polices [13, 24, 37, 56]. A referential game usually consists of a speaker seeing a target object and sending a message about it, and a listener, who needs to guess which is the target the speaker sees from a set of candidate objects. It is widely used in the study of emergent languages, and communication is the main focus of the task. Another line of research learns a multi-agent cooperative task with or without a communication channel to see if adding this channel will enhance the performance.

In addition to demonstrating successful communication in different scenarios, one of the human language properties, compositionality, is often studied in the structure of the resulting communication protocols [1, 45, 53]. Many works [31, 33, 34] have illustrated that compositionality can be encouraged by environmental pressures. Kottur *et al.* [33] show that under a series of specific environmental pressures, agents can be "coaxed" to communicate the compositional atoms each turn independently in a grounded multi-turn dialog. More general impact of environmental pressures on compositionality are investigated [7, 34], including different vocabulary sizes, different message lengths, carefully constructed distractors, etc.

Compositionality enables languages to represent complex meanings using meanings of its components [14]. It preserves the expressivity and compress-

ibility of the language at the same time [31]. Moreover, often emergent languages can be hard to interpret for humans. With the compressibility of fewer rules underlying a language, emergent languages can be easier to understand by humans. Intuitively, a language that is more easily understood should also be easier-to-teach as well.

This thesis focuses on investigating the structural properties of the resulting protocols/languages, and how they are influenced by the environment under a new pressure: ease-of-teaching. Specifically, we explore the connection between ease-of-teaching and the structure of the language through empirical experiments using simple referential games. Firstly, we show that a compositional language is, in fact, easier to teach than a less structured one, given the same agent architecture.

Secondly, to facilitate the emergence of easier-to-teach languages, we create a new environmental pressure for the speaking agent, by periodically forcing it to interact with new listeners during training. Since the speaking agent needs to communicate with new listeners over and over again, it has an encouragement to create a language that is easier-to-teach. We explore this idea, introducing new listeners periodically to replace old ones, and measure the impact of the new training regime on the ease-of-teaching of the language and its structure. We show that our proposed reset training regime, not only results in easier-to-teach languages, but also that the resulting languages become more structured over time.

Thirdly, the experiments sequentially introducing new agents suggest that the key environmental pressure — ease-of-teaching — may come from learning with a population of other agents (e.g., Jaderberg *et al.* [22]). Besides that, an explicit (and large) population can also smooth out abrupt changes to the training objective when new learners are added to the population. However, in a second set of experiments we show that these advantages surprisingly actually remove the pressure to the speaker. In fact, just the opposite happens: more abrupt sequential changes appear to be key in increasing the entropy of the speaker's policy, which seems to be leading to increasingly structured, easier-to-teach languages.

In summary, the main contributions of this thesis include:

1. A connection between ease-of-teaching and compositionality, thus introducing ease-of-teaching as a factor to evaluate emergent languages;

2. A training regime of repeatedly teaching new listeners as a pressure to increase ease-of-teaching (and compositionality) of the languages; and

3. A demonstration that incorporating new listeners abruptly instead of smoothly is key to increasing the effect.

The structure of the thesis is arranged as follows: Chapter 2 gives a brief review of the training algorithm we use and the related works people have done in the emergent communication field. Chapter 3 introduces our experimental setup, including the game context, agent architectures, training method and evaluation metrics. Chapter 4 proposes that there is a connection between ease-of-teaching and compositionality of the emergent languages and how incorporating a new listener sequentially into the training regime increases both. Chapter 5 explores whether a population of listeners, which smooths out the abrupt changes, could increase the pressure. Chapter 6 discusses how different hyperparameters affect different training regimes and further explores the impact of hyperparameters in a harder setting, with a binary vocabulary. The conclusions we find and future work are included in Chapter 7.

Chapter 2

Background and Related Works

2.1 Reinforcement Learning

Reinforcement Learning (RL) is goal-directed learning through interacting with the environment [58]. Recently, reinforcement learning combined with advances in deep neural networks [36] has achieved notable success in classic Atari 2600 games [44], mastering the game of Go [50] and robotics [38].

2.1.1 Basic Concepts

The problem of RL is typically formulated as a finite Markov Decision Process (MDP). A finite MDP is defined as a 4-tuple ($\mathcal{S}$, $\mathcal{A}$, p, r), where

- $\mathcal{S}$ is a finite set of states,
- $\mathcal{A}$ is a finite set of actions,
- p is the state transition probability, $p(s'|s,a) = Pr(S_{t+1} = s'|S_t = s, A_t = a)$, and
- r is the expected immediate reward received after transiting from s to s' under action a.

The agent is the decision maker or learner. It interacts with the environment in a sequence of discrete time steps. At each time step t, the agent receives a representation of the environment state S_t, and selects an action A_t based on S_t. Then at the next step, the agent receives a reward R_{t+1} from

the environment which transits to a new state S_{t+1}. The state transition of an MDP satisfies the Markov property (i.e., the next state S_{t+1} conditions only on S_t and A_t, not any previous states and actions).

The goal of an RL agent is to maximize the cumulative reward received over time, which is called the return G_t.

$$G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \tag{2.1}$$

where discount rate γ is a parameter, $0 \leq \gamma \leq 1$.

A policy $\pi(a|s)$ is a mapping from states to actions, specifying the probabilities of the actions the agent selects in state s. Value functions are used to decide how good a state or a state-action pair is under policy π. The value function of a state s under a policy π is defined as

$$v_\pi(s) = \mathbb{E}_\pi[G_t|S_t = s] = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} | S_t = s \right] \tag{2.2}$$

Similarly, the value function of taking an action a in state s under a policy π is defined as

$$q_\pi(s, a) = \mathbb{E}_\pi[G_t|S_t = s, A_t = a] = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} | S_t = s, A_t = a \right] \tag{2.3}$$

$v_\pi(s)$ and $q_\pi(s, a)$ are the state-value function and action-value function for policy π, respectively. Importantly, $v_\pi(s)$ and $q_\pi(s, a)$ can also be defined recursively through the Bellman equation [3]. Many algorithms are derived from this recursive definition, giving the family of temporal-difference (TD) learning methods [57]. We do not dive it into details here since most of the methods in this thesis do not rely on this.

Finding a policy that maximizes the expected return is the key problem in RL. One way to obtain a policy is from estimated action-value functions or state-value functions (e.g., using ϵ-greedy), and then using the policy to gather data to improve the estimated value functions. These are called value-based methods. For example, the class of Q-learning methods usually fall into this category. Another approach is to parametrize the policy and directly optimize the policy parameters, which are called policy gradient methods. My work and most other works mentioned in this thesis use policy gradient methods.

2.1.2 Policy Gradient Methods

Policy gradient methods directly parametrize $\pi(a|s) = Pr(a|s, \theta)$ and update the policy parameters in the direction of an estimate of the gradient of the objective. The policy objective function is defined differently in episodic tasks with finite time steps, versus continuing tasks of infinite horizon. In episodic cases, the policy objective function is typically defined as the value of the start state

$$J(\theta) = v_{\pi_\theta}(s_0) = \sum_s \mu_{\pi_\theta}(s) v_{\pi_\theta}(s) = \sum_s \mu_{\pi_\theta}(s) \sum_a \pi_\theta(a|s) q_{\pi_\theta}(s, a) \tag{2.4}$$

where μ_{π_θ} is the stationary distribution of the Markov chain for π_θ. In continuing tasks, we can define the objective in terms of the average reward per time step.

$$J(\theta) = \sum_s \mu_{\pi_\theta}(s) \sum_a \pi_\theta(a|s) \sum_{s',r} p(s', r|s, a) r \tag{2.5}$$

Computing $\nabla_\theta J(\theta)$ is tricky since it depends on the unknown state distribution following π_θ starting at s_0. Thanks to Policy Gradient Theorem [59], it follows that

$$\begin{aligned} \nabla_\theta J(\theta) &\propto \sum_s \mu_{\pi_\theta}(s) \sum_a q_{\pi_\theta}(s, a) \nabla_\theta \pi_\theta(a|s) \\ &= \sum_s \mu_{\pi_\theta}(s) \sum_a \pi_\theta(a|s) q_{\pi_\theta}(s, a) \frac{\nabla_\theta \pi_\theta(a|s)}{\pi_\theta(a|s)} \\ &= \mathbb{E}_{\pi_\theta}[q_{\pi_\theta}(S_t, A_t) \nabla \ln \pi_\theta(A_t|S_t)] \end{aligned} \tag{2.6}$$

Therefore, the gradient of the policy objective does not include the gradient of the state distribution with respect to the policy parameters.

One of the classical policy gradient methods is REINFORCE [61] algorithm. It uses the empirical discounted return $\gamma^t G_t$ as an unbiased sample of $q_{\pi_\theta}(S_t, A_t)$ to update the policy parameters.

$$\theta_{t+1} = \theta_t + \alpha \gamma^t G_t \nabla_\theta \ln \pi_\theta(A_t|S_t) \tag{2.7}$$

where α is the the algorithm's step size.

Advantages of policy gradient methods include being capable of learning stochastic policies, and being effective in high-dimensional and continuous action spaces. They also work well in POMDPs (Partial Observable Markov

Decision Process) [25]. In a POMDP, an agent cannot directly observe the underlying state, instead it receives an observation $o \in \Omega$ with conditional observation probability $O(o|s', a)$. In this case,

$$\nabla_\theta J(\theta) = \mathbb{E}_{\pi_\theta}[q_{\pi_\theta}(O_t, A_t) \nabla \ln \pi_\theta(A_t|O_t)] \tag{2.8}$$

One of the disadvantages of policy gradient methods is that the estimates of policy gradients can have high variance. To reduce the variance, a baseline value can be subtracted from the return without introducing bias [58]. A common choice for the baseline is an estimate of the state value. Another approach is to combine a critic to estimate the action-value function, together with the policy action selector (actor), resulting in the family *actor-critic* methods [32].

These methods are also often slow to converge and sensitive to choose the learning rate. Therefore, an adaptive step-size stochastic gradient descent optimization algorithm, like Adam [27] is commonly used in practice.

2.2 Communication in Reinforcement Learning

Successful communication requires a sender giving out a signal contingent on its private information (i.e., only known to itself), and a receiver interpreting it correctly. Communication can be in different forms, verbal or non-verbal, sometimes with a cost considered. In environments involving multiple agents, communication can be important in many different types of tasks — cooperative, competitive, or mixed cooperative-competitive — by achieving coordination [24], covert/deception [41] and negotiation [6]. Fully cooperative tasks are the most investigated setting, where all the agents share the goal of maximizing the same expected return. Partially observable environments are also usually considered to stress the necessity of communication. Challenging multi-agent benchmarks that would benefit from agents having communication capabilities, have been purposed, including Hanabi [2]. Real world applications like self-driving cars could also directly benefit from communication capabilities.

Previous studies on multi-agent problems involving communication mostly use predefined protocols, with a few exceptions using tabular RL approaches

[16, 26, 60] or an evolutionary algorithm [15] to *learn* to communicate. Recently, there have been a revival of research in automatically learning the communication protocols using deep reinforcement learning. The field of developing communication protocols/strategies among agents without pre-defining them has been dubbed *emergent communication*. Recent research on using deep reinforcement learning in this field can be roughly divided into two lines, depending on whether communication actions are the only action the agent takes. One line is adding a communication channel alongside the behaviour policy when communication does not directly affect the environment [13, 45, 56]. The goal is to enhance the behaviour performance via communication. In the other line, achieving communication itself is the main goal, and thus the emerged communication protocols are the main concern [7, 33, 34]. Related works are introduced separately in the following based on this categorization.

The study of emergent communication is important for a number of reasons. Agents with communication capabilities can enhance their performance in multi-agent tasks. Explicitly learning communication by developing a language/protocol can be beneficial when agents need to coordinate with humans or making decisions that are more interpretable to humans [45]. It also provides a way to learn a grounding of language in a pragmatics perspective [20], in contrast to modelling the statistical patterns of natural languages. It is also in the scientific interest to provide insights to how the structure in the human languages are originated and evolved.

2.2.1 RIAL and DIAL

In the first attempt at learning communication protocols with deep learning approaches [13], two approaches were proposed, Reinforced-Inter-Agent Learning (RIAL) and Differential Inter-Agent Learning (DIAL). Fully cooperative partial observable environments are considered. At each time step t, each agent a selects an environment action $u \in U$ that directly affects the environment and a communication action $m \in M$ observed by the other agents at the next step, based on its private observation o_t^a correlated with s_t.

In RIAL, behaviour actions and communication actions are separately

learned with two deep recurrent Q-networks (DRQN) [18], which replaces a feed-forward network approximating $Q(s, u)$ in deep Q-networks (DQN) [44] with a recurrent network $Q(o_t, h_{t-1}, u)$ for partial observable settings. u and m are selected separately from Q_u and Q_m by the action selector using ϵ-greedy. Experience replay is disabled since the non-stationarity of the environment can make the experience obsolete and misleading. Each agent is learned with independent Q-learning, where each agent independently and simultaneously learns its own Q-function. Learning and execution are the same and both decentralized. A parameter sharing option is also explored, where all the agents share the same two Q-functions Q_u and Q_m. In this case, learning becomes centralized since all the agents share the same value functions. Communication actions m in RIAL are all discrete messages.

DIAL exploits the advantage of centralized learning. During learning, it lets the gradients push back through the communication channel, which makes the communication messages real-valued. Only u is selected by the action selector, while the real-valued messages bypasses it to a discretise/regularise unit. This unit regularises the messages during centralized learning, while discretising it during decentralized execution. DIAL uses discrete protocols in their experiments, but naturally handles continuous protocols as well.

The main difference between RIAL and DIAL is that DIAL passes messages in continuous values during learning and lets gradients flow across agents, from the receiver to the sender to get richer feedback. While RIAL is end-to-end trainable within each agent, DIAL is end-to-end trainable across all agents.

2.2.2 Works on Enhancing Behavior Policy with Communication

We introduce seminal works on enhancing behavior policy with an explicit communication channel in this section.

Sukhbaatar *et al.* [56] also explore continuous communication protocols. They introduce CommNet, that coordinates agents learn to communicate before taking actions. CommNet is a deep feed-forward neural network that maps observations to actions for all agents. It can be trained via backprop-

agation and thus can be combined with reinforcement learning or supervised learning. Each layer is considered one communication step, which calculates a communication vector c and propagates a hidden state vector h for each agent. Each agent j is modeled as a multi-layer neural network, which at each time step t takes a hidden state vector h_t^j and a communication vector c^j and outputs h_{t+1}^j. Communication vectors c^j are the mean of all the other agents' hidden states. All agents share the same parameters. Though using continuous protocols, examining the protocols learned with CommNet in one of the tasks shows that a sparse communication protocol is learned that conveys meaningful information between agents.

A grounded communication environment (i.e., a physically-simulated world with multiple landmarks) is proposed and a basic compositional language is emergent with the following learning methods [45]. This paper models an end-to-end differential model of all agents sharing the same policy, similar to CommNet and DIAL/RIAL with parameter sharing. In addition, it learns a model of the environment state transition dynamics. Discrete communication symbols are sampled from a categorical distribution at test time. To make it differentiable during training, a Gumbel-Softmax distribution [17, 42] is used, which is a continuous relaxation of a discrete categorical distribution. An auxiliary prediction reward [11, 49] is used for predicting the other agent's goal to help policy training avoid local minima. There are some investigations into how variation in environments affects the communication strategies that arise, which we will revisit in the next section.

Jaques *et al.* [24] show that rewarding agents for having causal influence over other agents' actions is effective for multi-agent communication. Causal influence is evaluated via counterfactual reasoning. At each time step, an agent simulates alternative actions it could have taken and computes their effect on the behaviour of other agents. Actions that lead to bigger changes in other agents' behaviour are considered influential and rewarded. They show that the influence reward is equivalent to rewarding agents for having mutual information between their actions, which results in more coordinated behaviour. Enhanced coordination results are shown in challenging social dilemma envi-

ronments with or without an explicit communication channel. Importantly, the influence rewards for all agents can be computed in a decentralized way by enabling agents to learn a model of other agents. This is an advantage over the centralized learning methods mentioned before, which are unable to learn diverse policies among agents.

The most widely used metric in this line of research for examining if communication is emerging is an increase in reward after adding a communication channel. Lowe *et al.* [40] examine a few intuitive existing metrics and show some of them can be misleading. The authors conduct an experiment that shows strong indicators of communication according to speaker consistency [24] (quantifying the degree of alignment between an agent's messages and its actions) and qualitative analysis, but the message does not affect the other agents' behaviour. It turns out that positive signaling does not necessarily lead to positive listening. In this case, the emergent messages seem to be redundant compared to the payoff matrix provided in their experiment and suggests the correlation between actions and messages emerge as a byproduct of optimization. In this paper, a family of metrics called Causal Influence of Communication (CIC) are proposed to detect positive listening, which measures the causal effect that one agent's message has on another agent's behaviour.

2.2.3 Related Works in Referential Games

Referential games are used widely in the emergent communication literature. An illustration of a referential game is shown in Figure 2.1. Most referential games are variants of the Lewis signaling game [39], which is used as an attempt to develop a theory of convention and meaning by understanding the equilibrium properties of the signaling game. Referential games are fully cooperative games with two players, requiring that a sender signals about its private state, and a receiver unaware of the state observes the signal and must take some action on it. For each state, there is a unique action that is preferred by both. There are often many Nash equilibria in this game [9, 47], where each player is making one's own best decision assuming others' decisions remain unchanged. A preferable one is when the sender sends a different signal in each

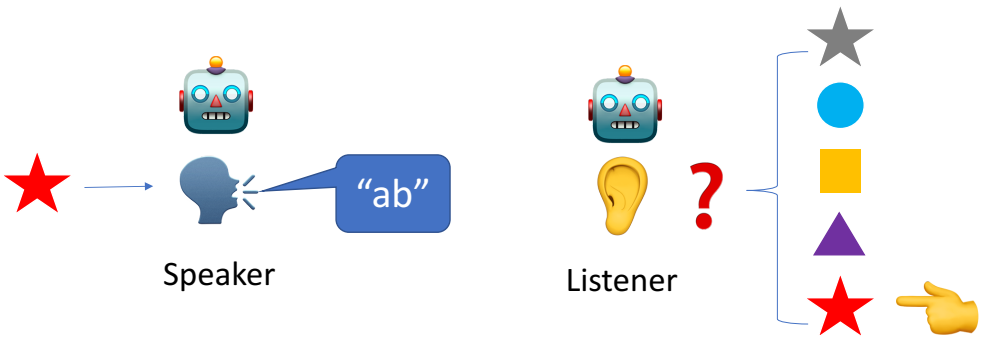

Figure 2.1: An illustration of a referential game.

state, and the receiver takes the appropriate action in every state. There are also pooling equilibria, where the sender sends the same signal in every state, and the receiver does its best without any additional information. Partial pooling equilibria exist when there are more than two states, signals, actions, where some information is conveyed and some are pooled [52].

Most works studying emergent communication in referential games are interested in what languages develop between agents [33, 34, 45]. Emergent languages that are more aligned with human languages may be preferred since they could be easier to be understood by humans. Moreover, emergent languages are learned in a grounded way, which means agents have mutual understanding about the task, could be preferred over capturing statistical patterns from a large corpora of human languages.

Recent works mostly model agents as deep neural networks and use REINFORCE as training methods so that a language consisting of discrete symbols can be developed naturally. Some perform a differentiable relaxation to enable end-to-end training. Discrete symbols are sampled in the forward pass, while a straight through Gumbel-softmax estimator [23] is used in the backward pass [19]. Choi *et al.* [7] propose an obverter technique, a differentiable learning algorithm for discrete communication. They assume others' minds are analogous to ours, thus maximizing the listener's understanding can be achieved by maximizing our own understanding. When the agent acts as a speaker, through introspection it sends the message most consistent with its

own observation. When it acts as a listener, parameters are updated via back-propagating the cross-entropy loss between its output and the true label (i.e., whether the agents are seeing the same image). Other than reinforcement learning techniques, Sirota [51] show that evolutionary search is viable for emergent communication, where neuroevolution is used to automate network design and find network weights of communicating agents.

Object representations in referential games are obtained differently as symbolic input [33, 34], pixel input processed with pre-trained neural networks [19, 34, 35], and pixel input trained together with the communication game [7]. Different forms of protocols are explored: messages of one symbol [35], a variable length sequence of symbols [7, 19], a dialog of a fixed number of turns [33], or even an adaptive length dialog [12].

A bidirectional multi-modal multi-step referential game is proposed, where the sender is exposed to visual modality and the receiver is exposed to textual description information [12]. Machine translation [37] can emerge from referential games when each agent speaks in their native language but share the same visual modality. Visual question answering and dialog agents can be fine-tuned to have better performance through playing cooperative image guessing games after supervised pretraining [10]. As an attempt to ground the agents' languages into human languages, combining the game with a supervised image labeling task encourages the protocols to be more interpretable [35].

2.3 Structure in Emergent Languages

An important hallmark in human language is compositionality, where the meaning of a complex expression is determined by its constituents [14]. There is no single mathematical definition of compositionality. Compositionality can be examined with qualitative measures [19, 34], quantitative measures [4, 34], or task success on held-out compositions of attributes not seen during the training [7, 33, 34, 45].

Environmental pressures play an important role in what kind of languages

emerge. This is also true when considering the origin and the evolution of human languages. Several works investigate the impact of environmental pressures on compositionality in different environments, including referential games [12, 34], a situated physical world consisting of multiple agents [45], and multi-turn dialogs [33]. The following key factors are considered in the current literature.

Maximum Message Length. Messages can end with an ending symbol in the vocabulary unless a maximum message length is achieved [12, 19, 34]. Different message lengths together with a same vocabulary size are explored [34]. With a shorter message length, the unique messages used are very few and show a high level of ambiguity (i.e., using the same message to represent different concepts). When the message length gets larger, the number of concepts per message are reduced and the communication accuracy gets higher. However, the resulting longer messages are more challenging to analyze for compositional patterns [12].

Vocabulary Size. A limited vocabulary is found to be essential to the emergence of compositionality in the task of a multi-turn dialog [33]. In their setting, objects have three attributes and each attribute has four possible values, where a minimum vocabulary of four is used. To have a compositional dialog actually emerge, a specific pressure needs to be combined with a minimum vocabulary in this work, removing the answering agent's memory each turn so that it can answer the value of the attribute being asked in that turn.

Inspired by Kottur *et al.* [33], the vocabulary size is set to be smaller than the size of objects [34]. A larger vocabulary is found to achieve high communication accuracy easily, while the languages appear less structured [12].

The emergence of compositionality requires the number of concepts to be a factor of vocabulary size [46]. While Mordatch *et al.* [45] tested a smaller vocabulary size they found the policy gets stuck in poor local minima where concepts became conflated. Instead they use a larger vocabulary size limit and a soft penalty function to prevent too many symbols from being used. The penalty is done by making more popular words survive with a rich-get-richer

dynamics. During training the larger vocabulary gives the policy optimization the chance to explore, but an active smaller vocabulary size is found during learning.

Carefully Constructed Data. Evtimova *et al.* [12] notice that randomly sampling object pairs with different colors and shapes let agents focus only on colors and ignore shapes. Note that in their experiment, the objects have 8 possible colors and 5 possible shapes. Thus, a carefully constructed mini-batch of data are presented, with mixed combinations of specific percentages of same and different colors and shapes.

The distractor selection process is also examined. A uniform selection of distractors is usually considered in the literature, but it does not reflect the context the object is in [34]. To tackle this, the authors experiment with a non-uniform selection of context-dependent distractors mimicking the normalized object co-occurrence statistics. This makes the ambiguous messages less affected by the visual context co-occurrences.

The number of distractors and the balance between the number of possible values for different attributes are also explored [34].

Physical Environment. The physically-simulated environment is conceived as a pressure contributing to the syntactic structure in the emergent languages [45]. The signal for a specific action always emerges first, since it takes time to accomplish the action in the environment.

In a simulated environment with objects of different colors and shapes, a strategy of communicating the absolute position of objects is developed [34]. Thus, different viewpoints can be involved in the experiment design to bias against directly communicating the absolute position.

2.4 Relevant Works in Cultural/Language Evolution

The idea of bringing uninformed learners into the learning process is investigated in cultural/language evolution field.

Human languages are culturally transmitted at least to some extent, where

children learn their languages by observing others' use of language [28]. Based on observational learning in cultural transmission, iterated learning [54], a paradigm for studying the origins and evolution of structure in human languages, has been studied for decades [29, 30]. In iterated learning, the output of a fully learned individual is the input for another uninformed one, and the language transmission between an individual and next is incomplete and partial. As a result of this repeated transmission bottleneck, compositional languages emerge due to stimulus poverty.

In iterated learning, language learning and language use are separate. Thus, the ease of language learning is solely connected to the compressibility of languages [31]. More compressed languages with simpler rules (e.g., a language with only one word for all objects) are easier to learn but quite ambiguous. Thus, compressibility and expressivity are seen together as competing pressures for compositional language to emerge.

Similar phenomenon have been observed in behavioral ecology. For example, in the pigeon route-navigation problem, partnering with an uninformed pigeon after several flights is found to improve homing efficacy over successive generations [48]. This is a real example that animals further progress by incorporating different members in the task and accumulating modifications from more than one individual.

Concurrently with our work, Cogswell *et al.* [8] explicitly incorporate cultural transmission into conversational agents modeled as deep neural networks. They try out different strategies to replace a questioning bot and an answering bot periodically in a single and multiple pair(s) of agents. Compositionality is evaluated in a generalization test and changes of languages between generations are examined to see if cultural transmission happens.

2.5 Role of the Thesis

In this thesis, we learn communication policies between multiple agents with partial observability using reinforcement learning. We adopt policy gradient methods, specifically REINFORCE, as our training method. We use discrete

symbols to communicate. Our training method is decentralized and does not involve any parameter sharing. We explore emergent languages in a pure communication environment without behaviour policies, i.e., referential games. We are interested in the structure of the emergent languages, and how the structure is influenced by a new environmental pressure of interacting with new listeners. Our proposed training regime of introducing new agents is connected to iterated learning in language evolution, although we do not explicitly create generations between agents since we have only one speaker and the agents learn only through interaction.

Chapter 3

Experimental Setup

We explore emergent communication in the context of a referential game, which is a multi-agent cooperative game and requires communication between a speaker S and a listener L. We first describe the setup of the game and then the agent architectures, training procedure, and evaluation metrics we use in this work.

3.1 Game Setup

The rules of the game are as follows. The speaker is presented with a target object $t \in O$ and sends a message m to a listener using a fixed-length ($l = 2$) sequence of symbols (m_1, m_2) from a fixed-sized vocabulary ($m_i \in V$ where $|V| = 8$). The listener is shown candidate objects $C = \{c_1, \ldots, c_5\} \subseteq O$ where $t \in C$ along with 4 randomly selected distractor objects, and must guess $\hat{t}$ which is the target object. If the listener guesses correctly $\hat{t} = t$, the players succeed and get rewarded $r = 1$, otherwise, they fail and get $r = 0$.

Each object in our game has a color and a shape. There are 8 colors (viz., black, blue, green, grey, pink, purple, red, yellow) and 4 shapes (viz., circle, square, star, triangle) in our setting, therefore $8 \times 4 = 32$ possible different objects. For simplicity, each object is represented by a 12-dimension vector concatenating a one-hot vector of color with a one-hot vector of shape, which is similar to work by Kottur *et al.* [33].

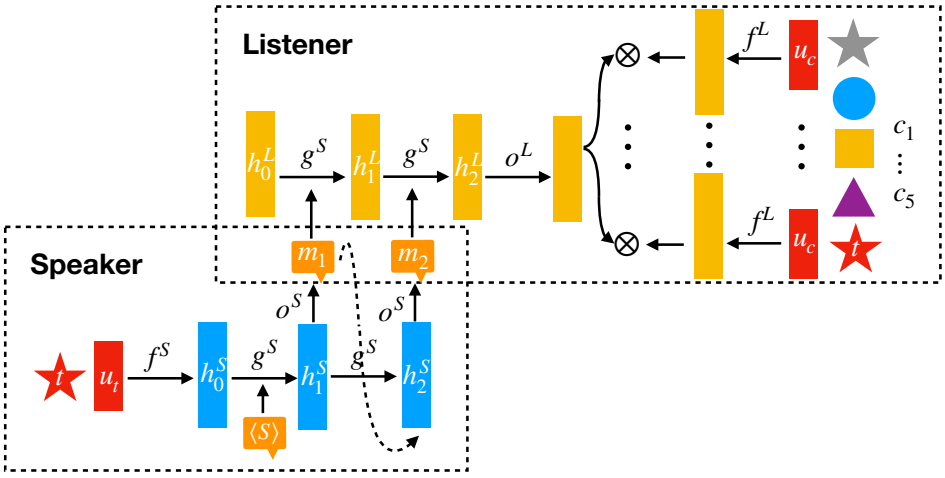

Figure 3.1: The architectures of the agents.

3.2 Agent Architecture

We model the agents' communication policies π^S and π^L with neural networks similar to related works [19, 34]. For the speaker, π^S stochastically outputs a message m given t. For the listener, π^L stochastically selects $\hat{t}$ given m and all the candidates C. In the following, we use θ^S to represent all the parameters in the speaker's policy π^S and θ^L for the listener's policy π^L. The architectures of the agents are shown in Figure 3.1.

Concretely, for the speaker, $f^S(t;\theta^S)$ obtains an embedding u_t of the target object t. This is processed by an LSTM [21] g^S to produce a message. At the first time step $\tau = 0$ of g^S, we initialize u_t as the start hidden state h_0^S and feed a start token $\langle S \rangle$ (viz., a zero vector) as the input of g^S. At the next step $\tau+1$, $o^S(h_{\tau+1}^S;\theta^S)$ performs a linear transformation from $h_{\tau+1}^S$ to the vocabulary space, and then applies a softmax function to get the probability distribution of uttering each symbol in the vocabulary. The next token $m_{\tau+1}$ is sampled from the probability distribution over the vocabulary and serves as additional input to g^S at the next time step $\tau+1$ until a fixed message length l is reached. For the listener, the tokens received from the speaker are input to an LSTM $g^L(m, h^L;\theta^L)$ and all the candidate objects are represented as embeddings u_c using $f^L(c;\theta^L)$. $o^L(h^L;\theta^L)$ transforms the last hidden state of g^L to an embedding and a dot product with $u_{c_1}, \ldots, u_{c_5}$ is performed. We then apply a

softmax function to the dot products to get the probability distribution for the predicted target $\hat{t}$. During evaluation, we use argmax instead of softmax for both S and L, which results in a deterministic language. The dimensionalities of the hidden states in both g^S and g^L are 100.

3.3 Training

In all of our experiments, we use stochastic gradient descent to train the agents. Our objective is to maximize the expected reward under the agents' policies $J(\theta^S, \theta^L) = \mathbb{E}_{\pi^S, \pi^L}[R(\hat{t}, t)]$. We compute the gradients of the objective by REINFORCE [61] and add an entropy regularization [43] term to the objective to maintain exploration in the policies:

$$\nabla_{\theta^S} J = \mathbb{E}_{\pi^S, \pi^L}[R(\hat{t}, t) \cdot \nabla_{\theta^S} \log \pi^S(m|t)] + \lambda^S \cdot \nabla_{\theta^S} H[\pi^S(m|t)]$$
$$\nabla_{\theta^L} J = \mathbb{E}_{\pi^S, \pi^L}[R(\hat{t}, t) \cdot \nabla_{\theta^L} \log \pi^L(\hat{t}|m, c)] + \lambda^L \cdot \nabla_{\theta^L} H[\pi^L(\hat{t}|m, c)]$$

where $\lambda^S, \lambda^L > 0$ are hyper-parameters and H is the entropy function, for the speaker $H = -\sum_m \pi(m|t) \log \pi(m|t)$.

For training, we use the Adam [27] optimizer with learning rate 0.001 for both S and L. We use a batch size of 100 to compute policy gradients. $\lambda^S = 0.1$ and $\lambda^L = 0.05$ are set in experiments in Chapter 4 and Chapter 5. Discussion about the effect of different λ^S and λ^L is included in Chapter 6.

Emergent language experiments in Chapter 4 and Chapter 5 are repeated 1000 times independently with the same random seeds for different regimes. Due to the computational constraints and that Chapter 6 mainly discusses the effect of different hyperparameters on the problem, experiments in Chapter 6 are repeated 100 times. In all the figures, the solid lines are the means and the shadings show a 95% confidence interval (i.e., 1.96 times the standard errors).

3.4 Evaluation

We evaluate the emergent languages in two ways, ease-of-teaching and the degree of compositionality of the language.

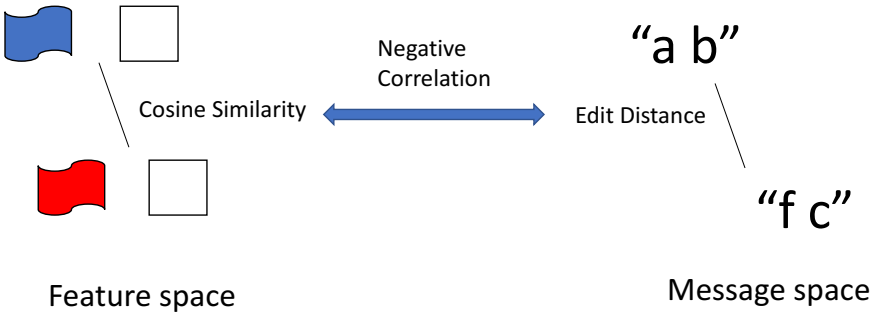

Figure 3.2: Topographic similarity is the negative correlation between distances in the feature space and in the message space of all object pairs.

To evaluate the ease-of-teaching of the resulting language (i.e., a new listener reaches higher accuracies with less training), we keep the speaker's parameters unchanged and produce a deterministic language with argmax, and then train 30 new randomly initialized listeners with the speaker for 1000 iterations and observe the ease-of-teaching of the languages.

There is not a definitive quantitative measure of language compositionality. However, a quantitative measure of message structure, i.e., topographic similarity, exists in the language evolution literature. Seeing emergent languages as a mapping between the meaning space and the signal space, topographic similarity is defined as the correlation between distances of pairs in the meaning space and those in the signal space [5].

We use topographic similarity to measure the structural properties of the emergent languages quantitatively [29, 34]. We compute this measure as follows: we exhaustively enumerate all target objects and the resulting messages from the deterministic π^S. We compute the cosine similarities s between all pairs of objects' vector representations and the edit distances d between all pairs of objects' messages. The topographic similarity is calculated as the negative Spearman ρ correlation between s and d. The higher the topographic similarity is, the higher the degree of compositionality in the language. Figure 3.2 shows how topographic similarity is computed.

Chapter 4

Experiments with Listener Reset

4.1 Compositionality and Ease-of-Teaching

In the introduction, we hypothesized that a compositional language is easier to teach than a less structured language. To test this, we construct two artificial languages, a perfect language with topographic similarity 1.0 and a permuted language. We create a perfect language by using 8 different symbols (a-h) from the vocabulary to describe 8 shapes, and choose 4 symbols (a-d) to represent 4 colors. The permuted language is formed by randomly permuting the mappings between messages and objects from the perfect language. It still represents all objects with distinctive messages, but each symbol in a message may not have a consistent meaning for shape/color. For example, 'aa' means 'red circle', 'ab' means 'red square', 'bb' means 'blue circle', and 'bc' means 'blue square'. An example of a perfect language and a permuted language are shown in 4.1 and 4.2. We then teach both languages to 30 randomly initialized listeners and observe on average how fast the languages can be taught. A language that is easier-to-teach than another, means reaching higher accuracies with less training.

We generated 1 perfect language and 100 randomly permuted languages, which had an average topographic similarity of 0.14. The training curves for both languages are plotted in Figure 4.3. We can see that the listener learns a compositional language much faster than a less structured language.

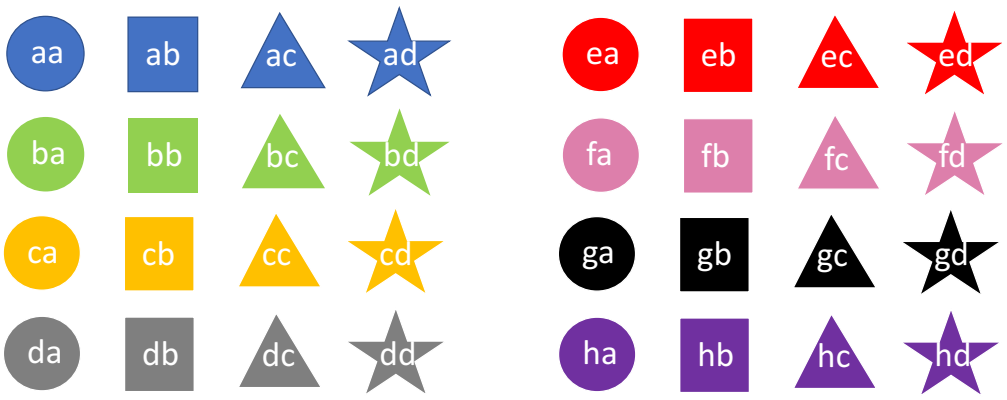

Figure 4.1: An example of a perfect language.

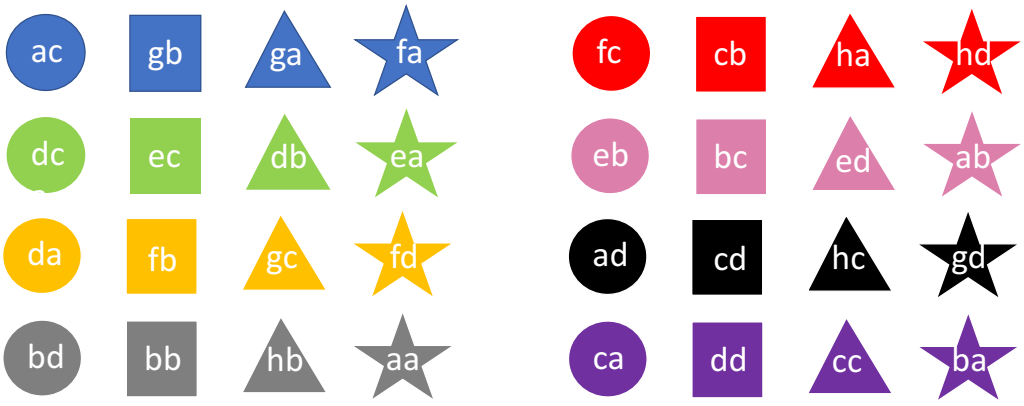

Figure 4.2: An example of a permuted language.

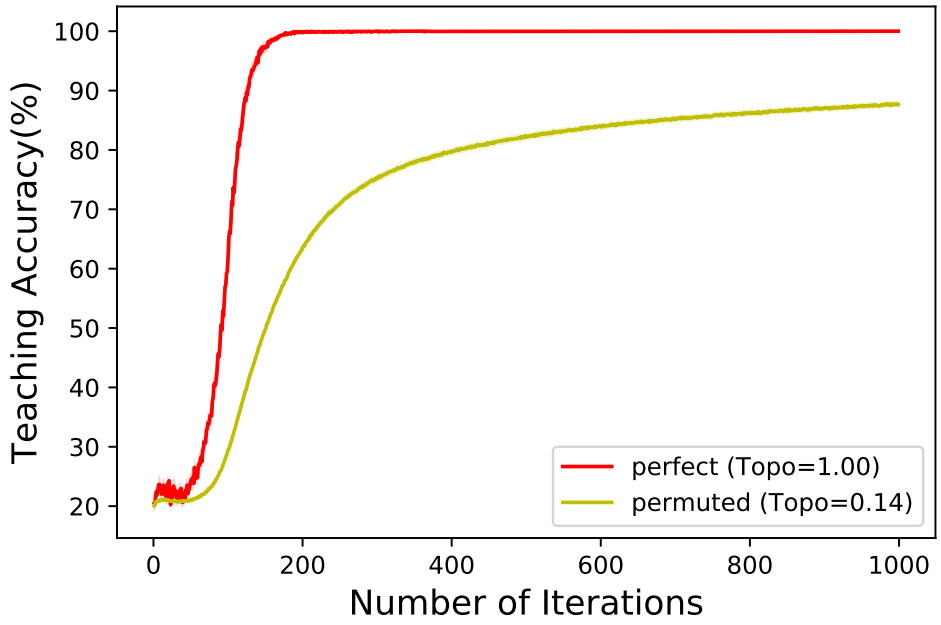

Figure 4.3: Ease-of-teaching of the artificial languages.

4.2 Experiments with Listener Reset

After observing a compositional language is easier to teach than a less structured language, we now explore a particular environmental pressure for encouraging the learned language to favour one that is easier to teach. The basic idea is simple, forcing the speaker to teach its language over and over again to new listeners.

To facilitate the emergence of languages that are easier to teach, we design a new training regime: after training a speaker S and a listener L to communicate for a fixed number of iterations, we reinitialize a new listener L' to replace the old one and have the speaker S and the new listener L' continue the training process. We repeat doing this replacement multiple times. We name this process "reset". Since the speaker needs to be able to communicate with a newly initialized listener periodically, this hopefully gives the speaker an environmental pressure to favour an easier-to-teach language.

We explore this idea by training a speaker with 50 listeners sequentially using the proposed reset regime and a baseline method with only one listener

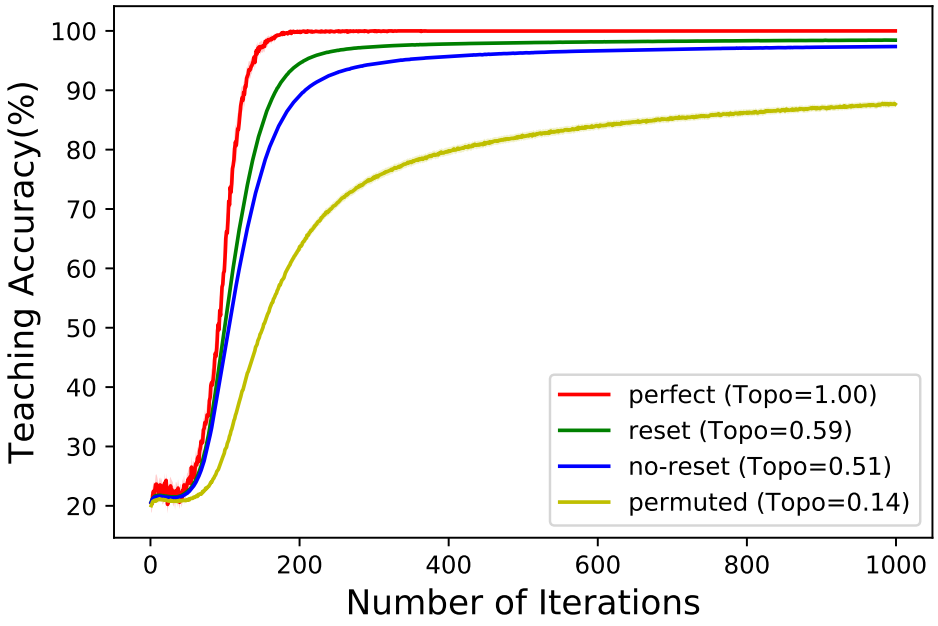

Figure 4.4: Ease-of-teaching of the emergent languages.

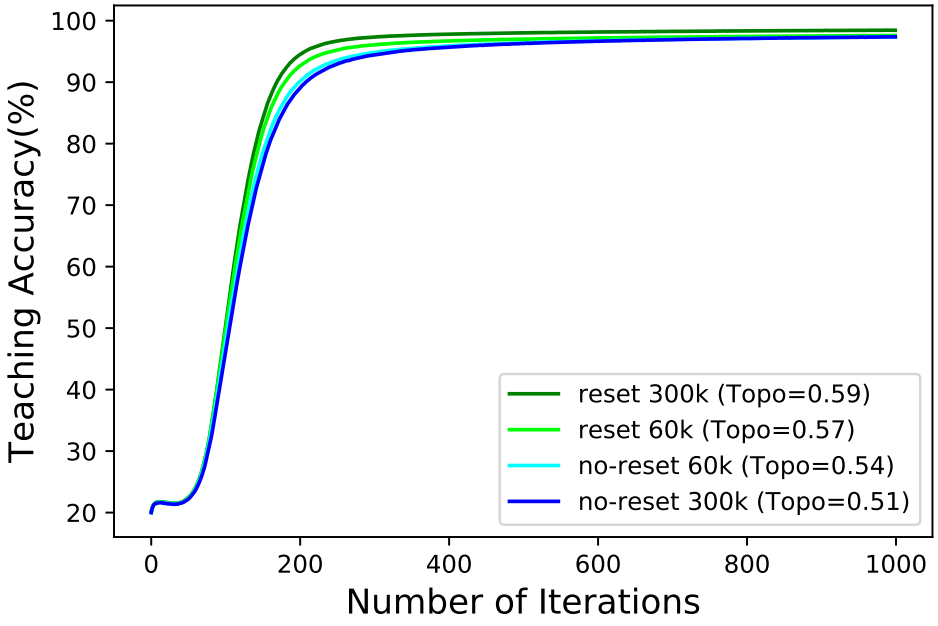

Figure 4.5: Ease-of-teaching of the emergent languages during training.

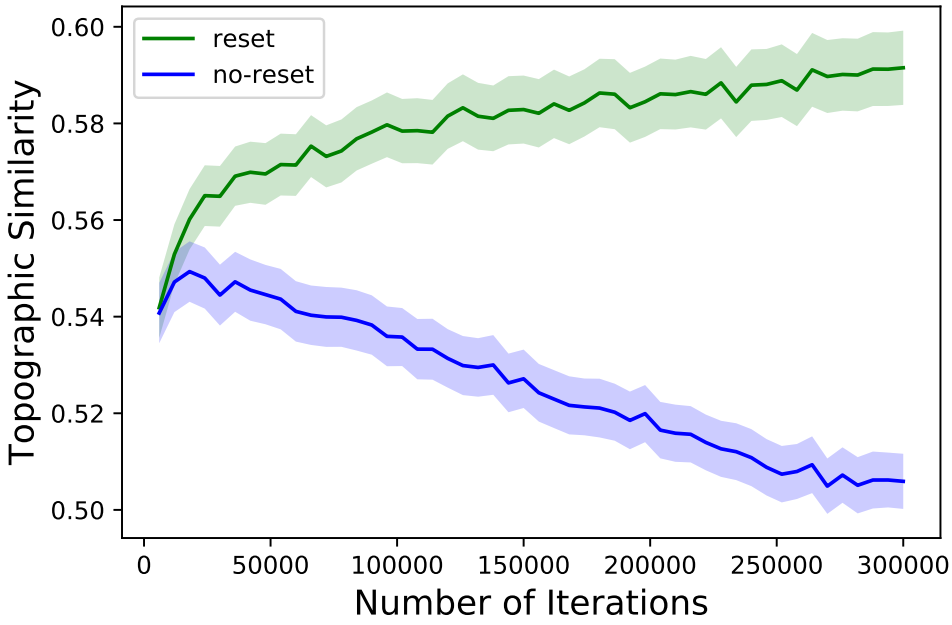

Figure 4.6: Comparison of topographic similarity under reset and no-reset regime.

for the same number of iterations. In the reset regime, the listener is trained with the speaker for 6k (i.e., 6,000) iterations and then we reinitialize the listener. Thus, the total number of training iterations is 6k $\times$ 50 = 300k. In the no-reset regime, we train a speaker with the same listener for 300k iterations.

4.2.1 Ease-of-teaching under the Reset Training Regime

After training, both methods can achieve a task success rate around 98% when teaching a new listener. In fact, after around 6k iterations agents can achieve high communication accuracy, but in this work we are interested in how language properties are affected by the training regime. Therefore, the following discussion is not about the communication success, but the differences between the emergent languages in terms of the ease of teaching and the degree of compositionality. We first evaluate the ease-of-teaching of the resulting language after 300k iterations of training and show the results in Figure 4.4. We can see that the languages emergent from the reset regime are on average easier

Table 4.1: A learned language with topographic similarity 0.81 from the reset regime

	black	blue	green	grey	pink	purple	red	yellow
circle	dh	bf	be	bb	bc	dg	da	dd
square	fh	ef	ee	eb	ec	eg	ea	ed
star	hh	hf	ce	cb	hc	hg	ha	hd
triangle	ah	gf	ge	gb	gh	gg	ga	gd

to teach than without resets.

We also test the ease-of-teaching of the emergent languages every 60k training iterations, to see the changes of ease-of-teaching during training. Results are shown in Figure 4.5, although for simplicity we are showing the ease of teaching for just one intermediate datapoint, viz., after 60k iterations. For the reset regime, the teaching speed of the language is increasing with training. For the no-reset regime, the teaching speed is in fact getting slower.

4.2.2 Structure of the Emergent Languages

But does the emergent language from the reset regime also have a higher degree of compositionality? We compute the topographic similarity of the emergent languages for both methods every 6k iterations (before the listener in the reset regime gets reset), and show how the topographic similarity evolves during training in Figure 4.6. In the reset regime, the topographic similarity rises with training to 0.59; while in the no-reset regime the metric drops with training to 0.51. This shows that not only are the languages getting easier to teach with additional resets, but the languages are getting more structured, although still not approaching a perfectly structured language.

Table 4.1 shows an example of one of the resulting languages from the reset regime. This is from an above average outcome where the topographic similarity is 0.81. In this example, each color is represented by a separate unique symbol as m_2 except that 'pink' reuses a different symbol 'h' once in 'triangle'. Each shape is represented by a disjoint set of symbols as m_1. {'b', 'd'} can represent 'circle', {'e', 'f'} can both mean 'square', {'c', 'h'} for 'star' and {'a', 'g'} for 'triangle'.

Table 4.2: A learned language with topographic similarity 0.71 from the no-reset regime

	black	blue	green	grey	pink	purple	red	yellow
circle	ff	fd	fa	fe	bd	fg	ff	fe
square	ea	hd	ha	ee	hd	eg	ea	ee
star	da	dd	dg	dh	dd	dg	da	dh
triangle	ga	gd	gg	ge	gd	gg	ga	ge

The most structured language from the no-reset regime after 300k iterations, with topographic similarity 0.71, is shown in Table 4.2. In this language, the first letter m_1 means shape, {'f', 'b'} for 'circle', {'e', 'h'} for 'square', 'd' for 'star', 'g' for 'triangle'. However, as for the color, same m_2 are shared by different meanings, making the language structured though ambiguous. 'black' and 'red' are not distinguishable, so are 'grey' and 'yellow'. 3 out of 4 'blue' and 'pink' objects are ambiguous except for 'circle'. 2 out of 4 'green' and 'purple' objects are ambiguous.

Chapter 5

Experiments with a Population of Listeners

We have so far shown introducing new listeners periodically creates a pressure for ease-of-teaching and more structure. One might expect that learning within an explicit population of diverse listeners (e.g., each having experienced a different number of training iterations) could increase this affect. Furthermore, one might expect a larger population could smooth abrupt changes to the training objective that occur when replacing a single listener. This is partly the role that experience replay plays in DQN to stabilize learning [44], and so we might see similar benefits.

5.1 Population Regime

We explore this alternative in our population training regime. We now have N listeners instead of 1, and each listener's lifetime is fixed to $L = 6k$ iterations, after which it is reset to a new randomly initialized listener. At the start, each listener is considered to have already experienced a different number of iterations, uniformly distributed between 0 and $L\frac{N-1}{N}$, inclusive — maintaining a diverse population of listeners with different amounts of experience. The speaker's output is given to all the listeners on each round. Each listener guessing the target correctly gets rewarded $R = 1$, otherwise $R = 0$. The speaker on each round gets the mean reward of all the listeners.

5.2 Experiments with Different Population Sizes

We experiment with different listener population sizes $N \in \{1, 2, 10\}$. Note that the sequential reset regime of the previous section is equivalent to the population regime with $N = 1$.

The mean teaching accuracy of the emergent languages for each regime is plotted in Figure 5.1. We can see that languages are easiest-to-teach from the reset regime, then a population regime with 2 listeners, then a population regime with 10 listeners, then the no-reset regime.

The average topographic similarity during training is shown in Figure 5.2. Topographic similarity from the population regime with 2 listeners rises to 0.57, but not as much as the reset regime. For the population regime with 10 listeners, the topographic similarity remains almost at the same level around 0.56.

From the results, we see that although larger populations have more diverse listeners and a less abruptly changing objective, this is not advantageous for the ease-of-teaching and the structuredness of the languages. Moreover, the population regime with a small number of listeners performs closer to the reset regime, while with a relatively large number of listeners seems closer to the no-reset regime.

5.3 Experiments with Resetting All Listeners

The sequential reset regime can be seen as resetting all listeners in a population with $N = 1$. We further experiment with resetting all listeners at the same time periodically and a baseline no-reset regime with a population of listeners $N \in \{2, 10\}$.

The mean teaching accuracy and topographic similarity of the language when $N = 2$ are plotted in Figure 5.3 and Figure 5.4. The ease-of-teaching curve of resetting all listeners overlaps with resetting 1 listener in the population. The languages from the no-reset regime are less easier-to-teach. While the topographic similarity rises to 0.59 when resetting all listeners periodically,

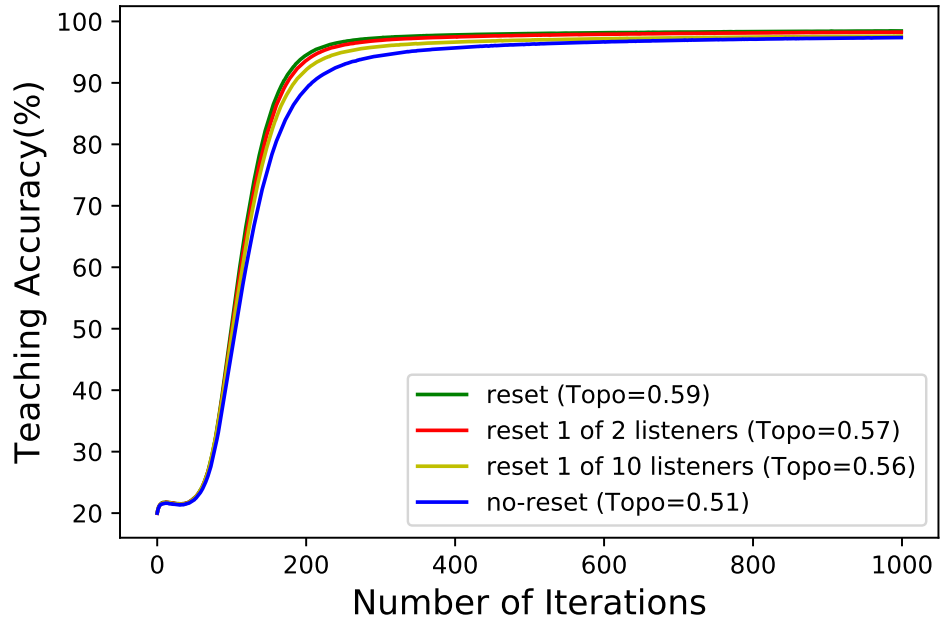

Figure 5.1: Ease-of-teaching of the languages under population regimes.

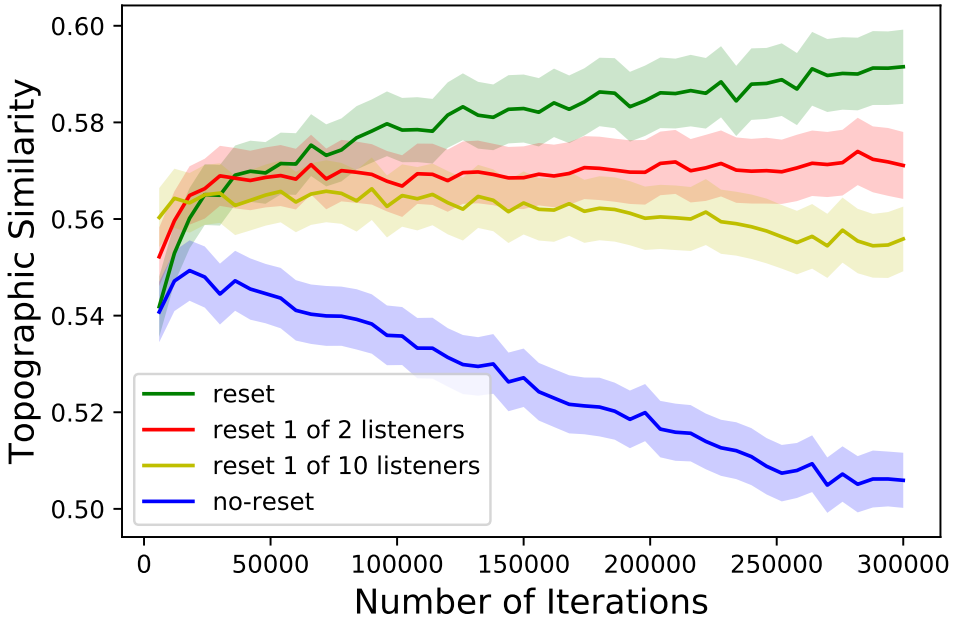

Figure 5.2: Comparison of topographic similarity under population regimes.

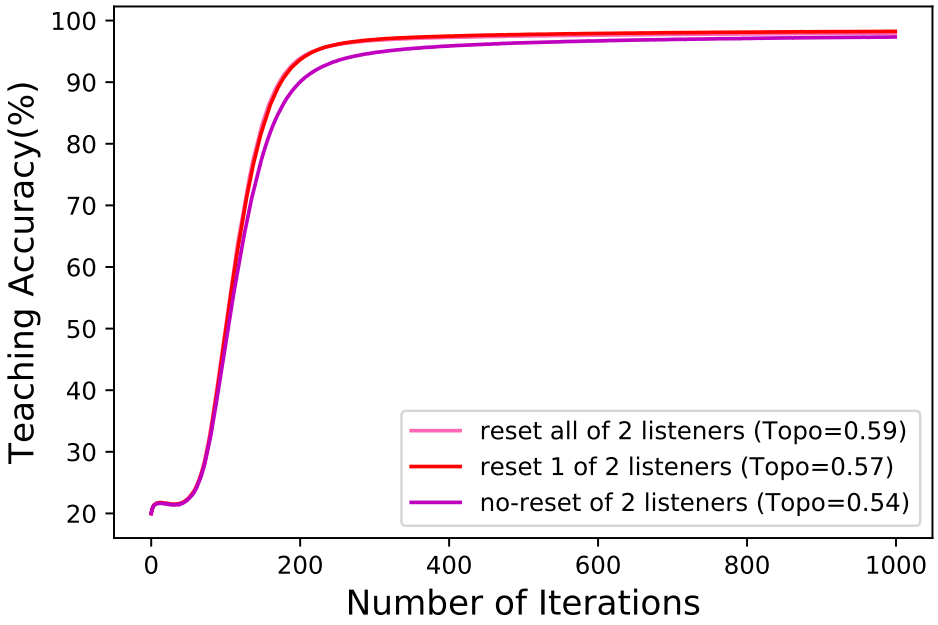

Figure 5.3: Ease-of-teaching of the languages when $N = 2$.

it decreases to 0.53 without resets. With a small population of 2 listeners, resetting all listeners and without resets show similar impact as $N = 1$.

As for a population of listeners $N = 10$, the mean teaching accuracy and topographic similarity of the languages are plotted in Figure 5.5 and Figure 5.6. Languages are easier-to-teach when resetting all listeners, then resetting 1 of 10 listeners, then no-reset in the population. And the gap between resetting all listeners and the no-reset regime in ease-of-teaching is smaller compared to $N = 2$. The topographic similarity rises to 0.59 when resetting all listeners, while it shows a similar trend of slight increase in resetting one of listeners and without resets.

Whenever $N = 1, 2, 10$, languages are easier-to-teach when resetting all listeners periodically than without resets. Moreover, the structuredness of the languages is the highest from resetting all listeners comparing to the others. Larger population of listeners are less likely to produce less easier-to-teach and less structured languages when no listener gets reset.

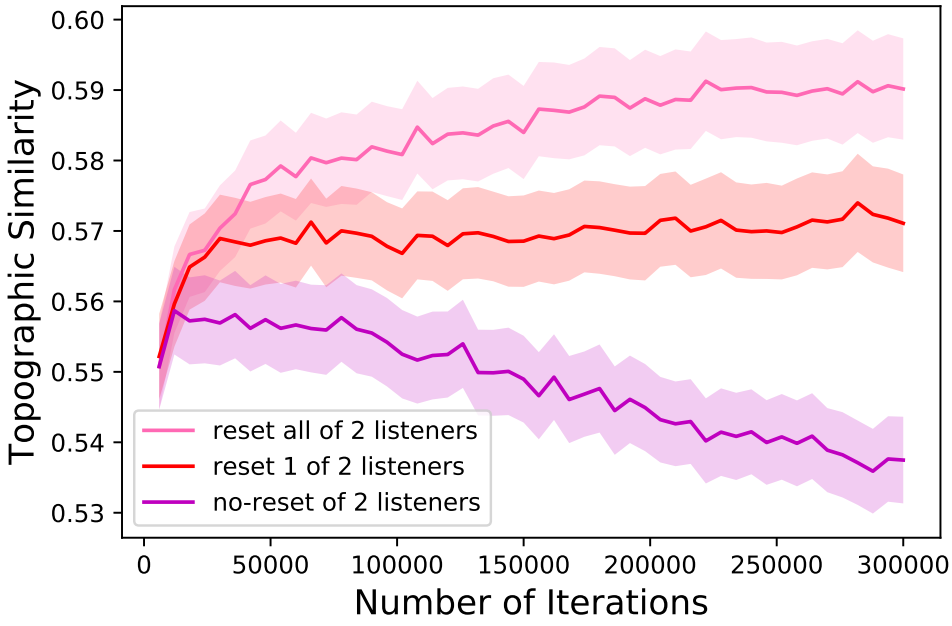

Figure 5.4: Comparison of topographic similarity when $N = 2$.

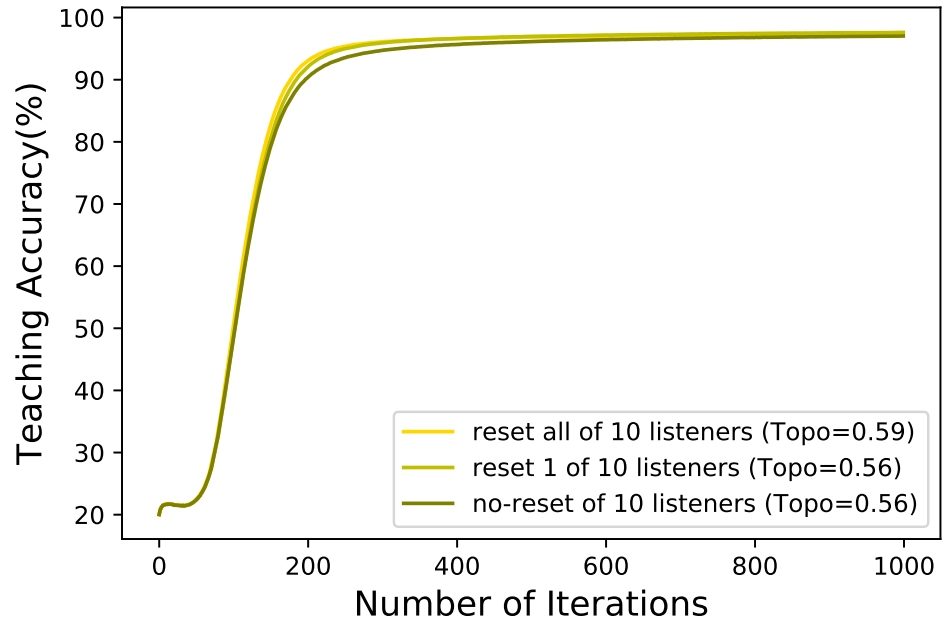

Figure 5.5: Ease-of-teaching of the languages when $N = 10$.

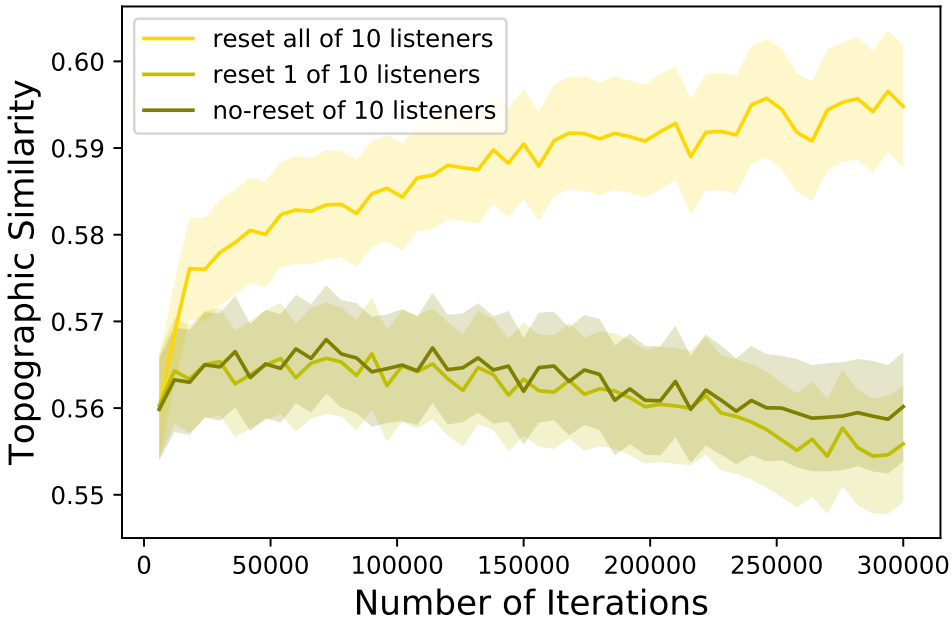

Figure 5.6: Comparison of topographic similarity when $N = 10$.

5.4 Discussion

In this section we investigate the behavior of the different regimes and give some thoughts on what might cause the emergent languages to be easier-to-teach and more structured.

To get a better view of what the training procedure looks like when resetting a single listener in different sizes of populations, we show the training curves of these regimes in Figure 5.7. In all the regimes, the speaker achieves communication success rate over 85% with listener(s) within 6k iterations. In the reset regime with $N = 1$, every 6k iterations the listener is reset to a new one, therefore the training accuracy drops down to 20%, a chance of a randomly guessing 1 target from 5 objects correctly. For a population of 2 listeners, every 3k iterations 1 of the 2 listeners gets reset, which makes the training accuracy drop down to around 55%. As for a population of 10 listeners, every 600 iterations 1 new listener jumps in the population while others still understand the current language, which makes the accuracy drop down to around 82%. For the no-reset regime, the agents maintain a high training

accuracy about 90% almost all the time.

When a new listener is introduced, the communication success is lower and the speaker gets less reward and benefits from increasing entropy in its policy due to the entropy term $H[\pi^S(m|t)]$ in the learning update. This explicitly created pressure for exploration may have the speaker unlearn some of the pre-built communication convention and vary the language to one that is learned more quickly. We plot the speaker's entropy during training in Figure 5.8, which backs up this explanation. We can see that there is an abrupt entropy change when a new listener is introduced to a population of 1 or 2 listener(s), which could possibly alter the language to be easier-to-teach. For the population regime with a large number of listeners, we cannot see abrupt changes in the entropy. Although 1 listener gets reset, the majority of the population maintain the communication with the speaker. Thus, the speaker is less likely to alter the communication language to one that the new listener is finding easier to learn.

This explanation also aligns with the result that resetting all listeners in different sizes of population will have a better performance. Since resetting all listeners will create a high abrupt entropy instead of being smoothed out by the others in the population when resetting one of the listeners. It would seem that the improvement comes from abrupt changes to the objective rather than smoothly incorporating new listeners.

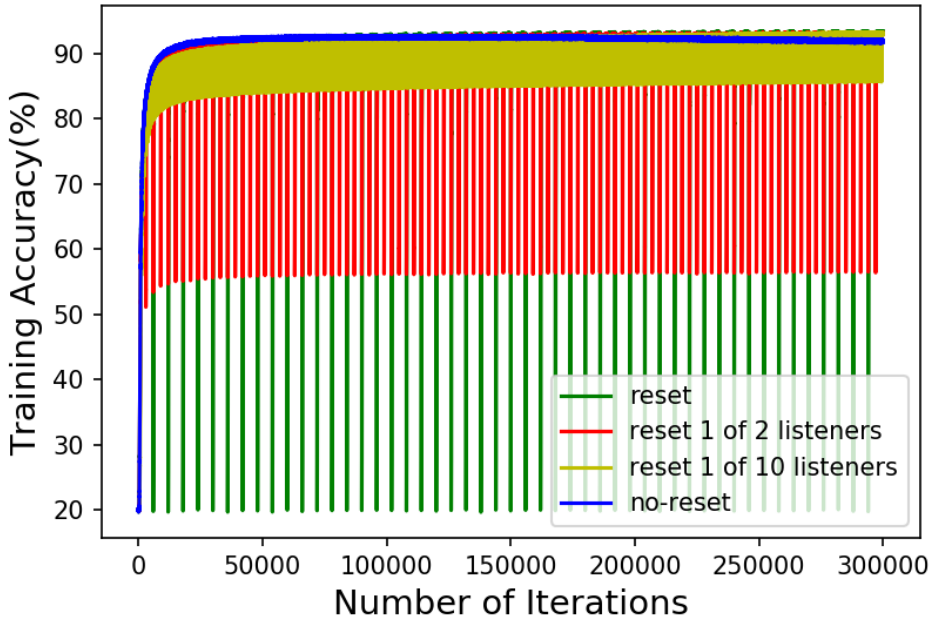

Figure 5.7: Training curves of the agents in different regimes.

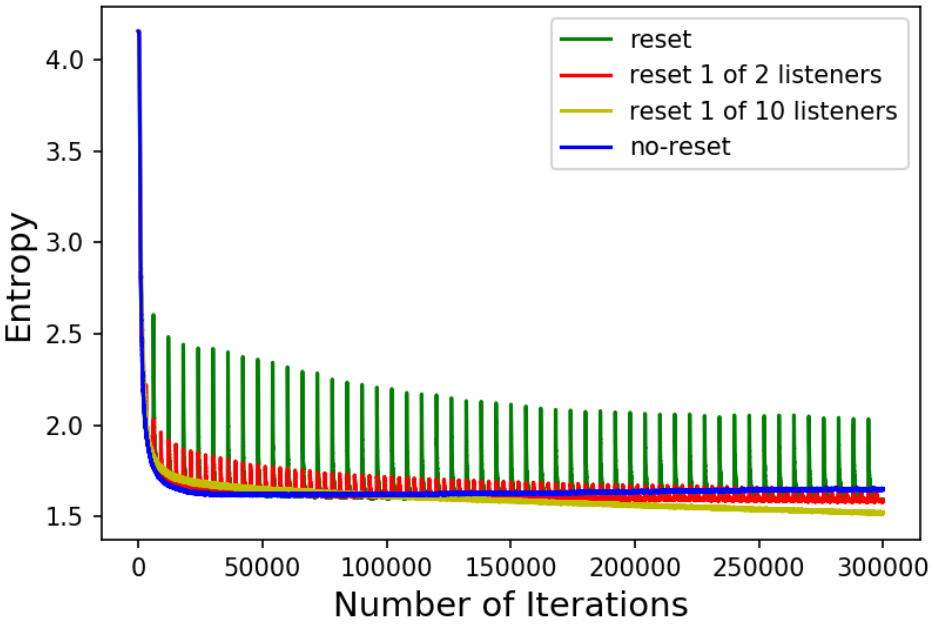

Figure 5.8: Speaker's entropy in different regimes.

Chapter 6

Further Experiments

Further experiments are presented in this chapter to understand how sensitive the 'reset' regime is to hyperparameters, and how it works in a harder setting.

6.1 Sensitivity to Hyperparameters

The previous chapters present some intriguing results with the same hyperparameter in different regimes. We discuss how sensitive the impact is with respect to different hyperparameters.

Successful communication between agents is often achieved relatively quickly and consistently when hyperparameters λ^S and λ^L (i.e., weights on the speaker's and the listener's entropy terms) are from $\{0.05, 0.1\}$. We run experiments of training 1 listener with or without resets using different combinations of λ^S and λ^L. We evaluate the ease-of-teaching and the topographic similarity of the languages after training using the same approach as before.

The ease-of-teaching curve of the languages with and without resets when $\lambda^S = 0.05$ and $\lambda^S = 0.1$, are shown respectively in Figure 6.1 and Figure 6.2. We find that with any hyperparameter combination the languages from the reset regime are easier-to-teach. However, when $\lambda^S = 0.05$ the curves with resets are closer to those without resets, compared to $\lambda^S = 0.1$. The impact of resets is smaller since exploration in the objective is weighted less. We can find some evidence for this by examining the speaker's entropy with different λ^S and λ^L in Figure 6.3 and Figure 6.4. We can see that when $\lambda^S = 0.05$ the entropy spikes are smaller, and so there is less opportunity for the policy to

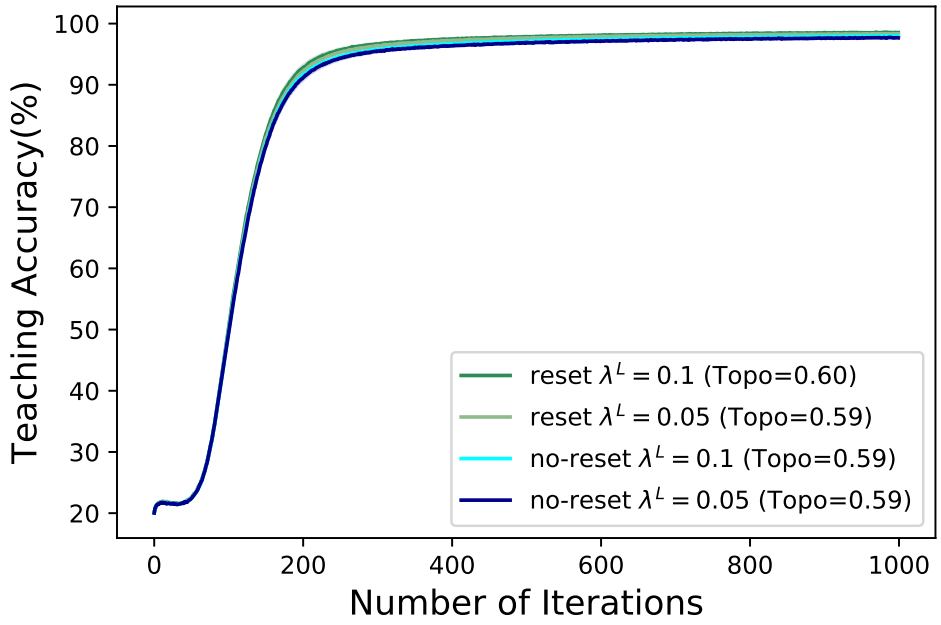

Figure 6.1: Ease-of-teaching of the languages when $\lambda^S = 0.05$.

explore.

Topographic similarity with different λ^S and λ^L are shown in Figure 6.5 and Figure 6.6. In both figures, with more resets topographic similarity rises at first, however, when $\lambda^S = 0.05$ the topographic similarity drops to the initial level eventually. We do not know why the measure goes down when $\lambda^S = 0.05$. Without resets, when $\lambda^S = 0.05$ topographic similarity stays at the same level, when $\lambda^S = 0.1$ it drops down.

We do not find much difference in performance between $\lambda^L = 0.05$ and $\lambda^L = 0.1$ in this setting, except that training deteriorates occasionally after 220k iterations without resets when $\lambda^S = 0.1$ and $\lambda^L = 0.1$.

6.2 Experiments with a Limited Vocabulary

We further experiment in a harder setting with a limited vocabulary (i.e., only '0', '1' characters). Are the emergent languages still easier-to-teach and more structured using the reset regime in this setting? Can we find "words" made up by multiple bits of "characters" in the language? In this section, we will

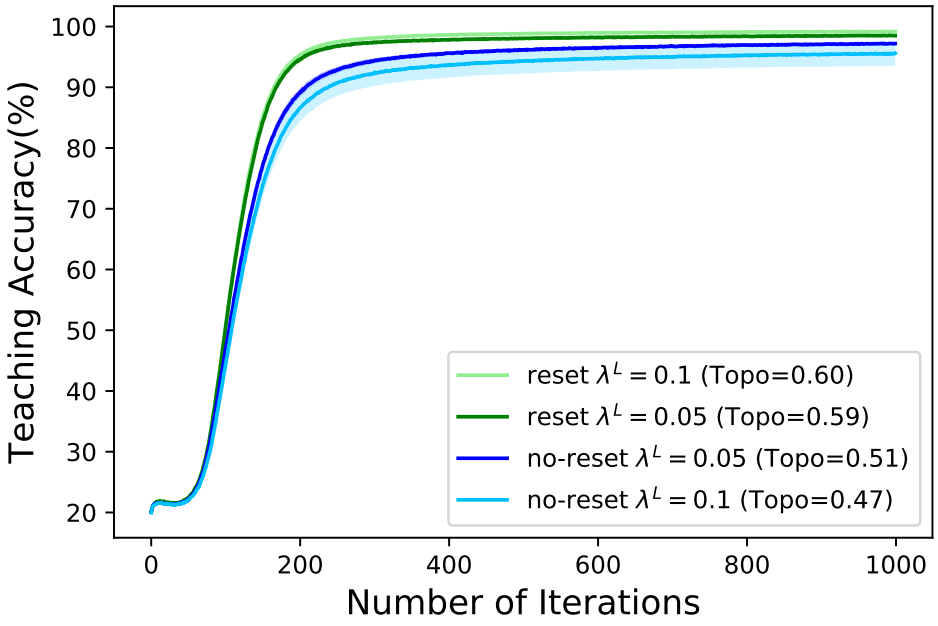

Figure 6.2: Ease-of-teaching of the languages when $\lambda^S = 0.1$.

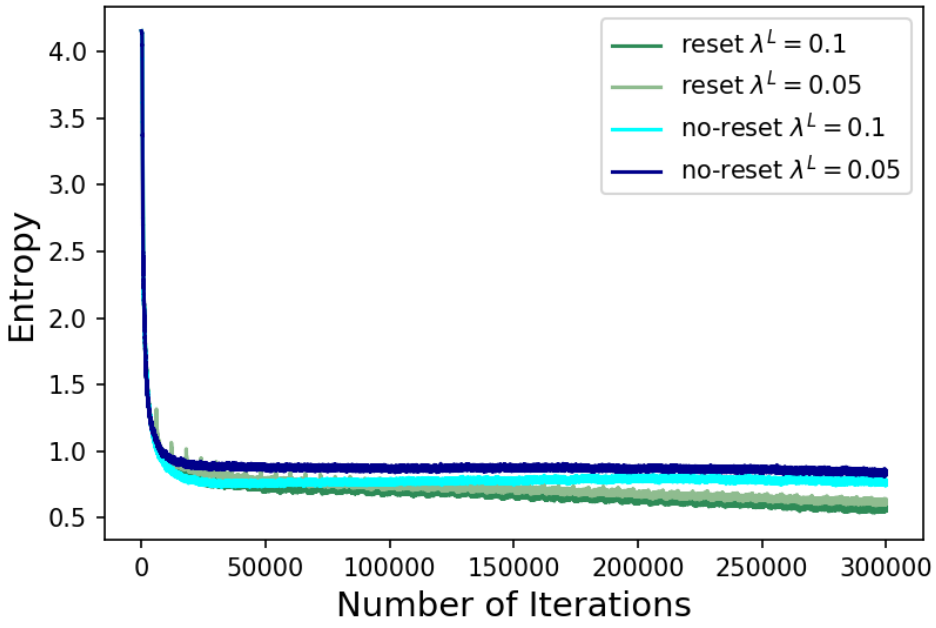

Figure 6.3: Speaker's entropy when $\lambda^S = 0.05$.

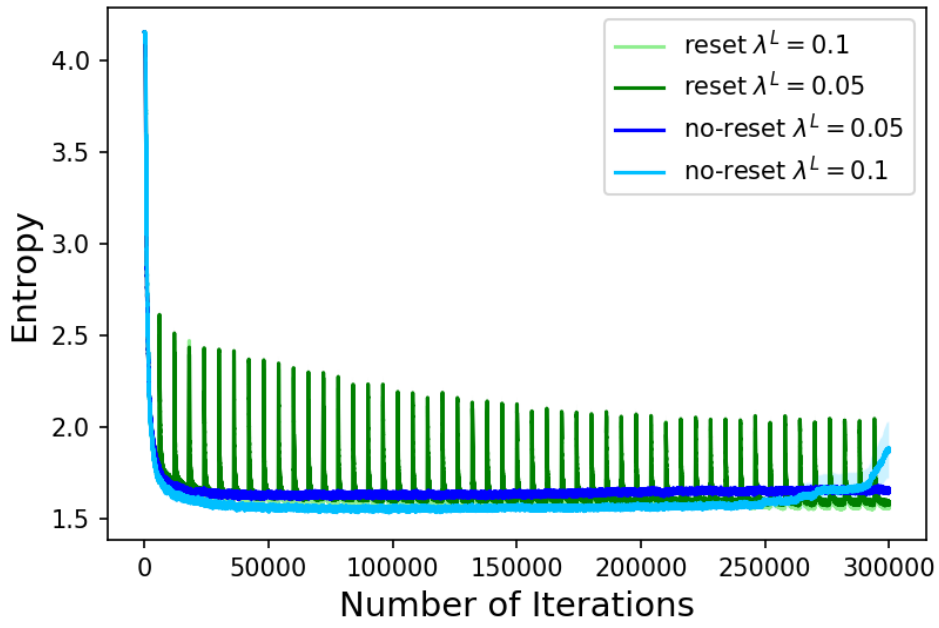

Figure 6.4: Speaker's entropy when $\lambda^S = 0.1$.

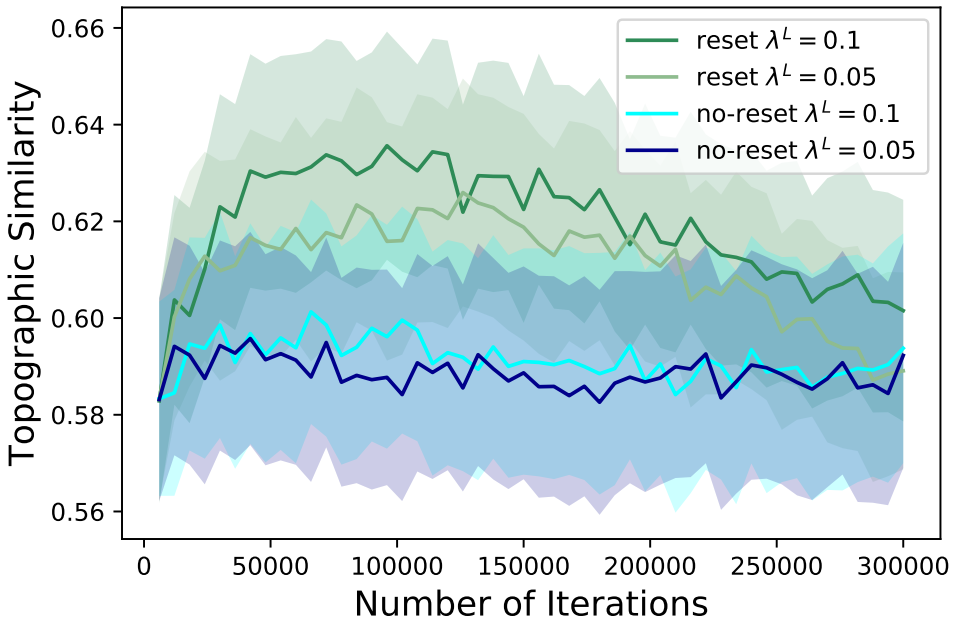

Figure 6.5: Comparison of topographic similarity when $\lambda^S = 0.05$.

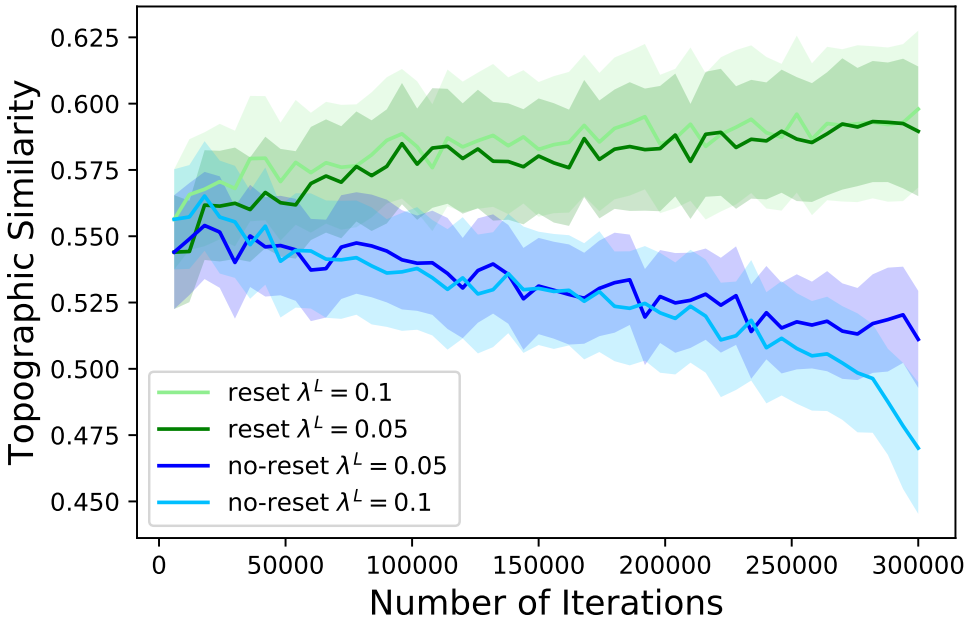

Figure 6.6: Comparison of topographic similarity when $\lambda^S = 0.1$.

explore these questions by reproducing the experiments with the reset and the no-reset regime in a binary vocabulary.

Since the vocabulary size is 2, the message length has to be at least 5 to be able to decode all 32 objects into different messages. We use message length $l = 5$ in the experiment. All the other settings stay the same as in the fourth chapter.

6.2.1 Compositionality and Ease-of-teaching

We first replicate the experiment whether compositionality and ease-of-teaching are related in the binary vocabulary setting. We compose two languages, a perfect language and a permuted language. The perfect language uses three bits to represent 8 colors and two bits for 4 shapes. The permuted language randomly permutes the mappings between messages and objects. We also experiment with constructing the perfect language in different ways, expressing shape first, color next (short as 'sc') or color first, shape next (short as 'cs'). We teach each differently constructed artificial language with 30 listen-

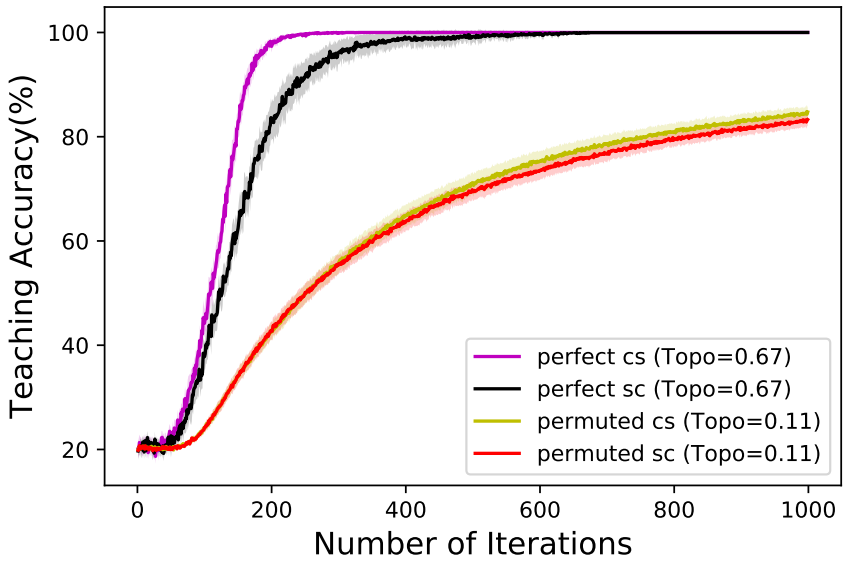

Figure 6.7: Ease-of-teaching of the artificial languages with a binary vocabulary.

ers independently for 1000 iterations. For permuted languages, results are also averaged over 20 random permutations. The training curve is shown in Figure 6.7. We can see that perfect languages are easier-to-teach than permuted ones. Expressing color first, shape next is slightly better than the other way around, although we do not have a good explanation for why this would be true. For simplicity, we only retain the results of 'cs' in the following figures as comparison.

6.2.2 Hyperparameters Search

Two hyperparameters need to be tuned, λ^S and λ^L before the entropy regularization term. We find hyperparameter search crucial in this setting since with some hyperparameters agents' strategies might converge to a (partial) pooling equilibrium. In these cases, the speaker finds a unitary language (i.e., uses the same message for all objects) or nearly unitary (i.e., uses the same message for all but one object). Thus, the communication success is low and we call such languages as degenerate languages [31].

We sweep λ^S and λ^L respectively over $\{0.1, 0.05, 0.02\}$ and train agents

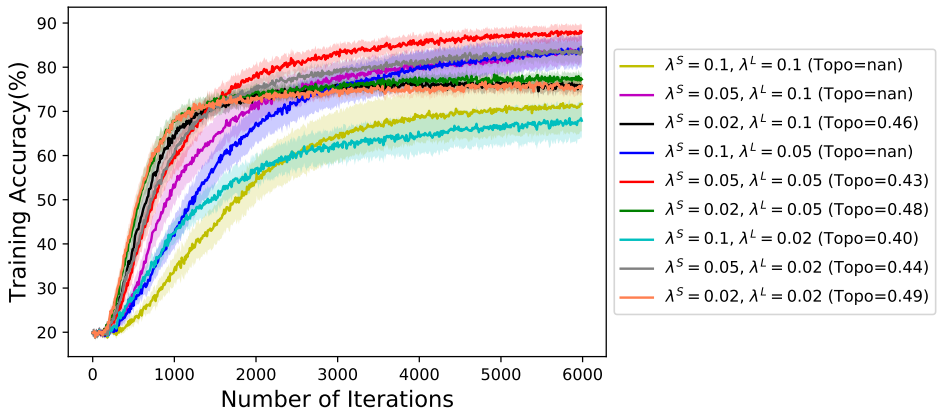

Figure 6.8: Hyperparameter sweep over λ^S and λ^L.

for 6k iterations without reset. Training curves of the 6k iterations with different hyperparameters are plotted in Figure 6.8. Each line is averaged over 100 randomly initialized trials with the same seeds. The average topographic similarity after training for 6k iterations is also shown in the legend. If any trial contains degenerate languages, the topographic similarity is 'nan'. Note that topographic similarity is undefined when there is a unitary language or a same message for all except one [29]. From this figure, since when $\lambda^S = 0.05$ and $\lambda^L = 0.05$ communication success is achieved fastest, we use those for the experiments.

When $\lambda^S = 0.05$ and $\lambda^L = 0.05$, we conduct the same experiment as before. For the reset regime, we swap out a new listener every 6k iterations for 50 times. For the no-reset regime, the speaker always talks to a same listener for 300k iterations. We evaluate the teaching speed of the language after 60k, 180k and 300k iterations by freezing the speaker's parameters and teaching the language to 30 new listeners. The teaching speed of languages learned in both regimes are in Figure 6.9. We also evaluate the topographic similarity measure every 6k iterations for both regimes and show in Figure 6.10. The teaching speed of the languages gets slow with training for both regimes, which corresponds to the decrease of the topographic similarity measure in both. For the no-reset regime, topographic similarity measure decreases fast at first and gradually slows down dropping. For the reset regime, the measure remains almost the

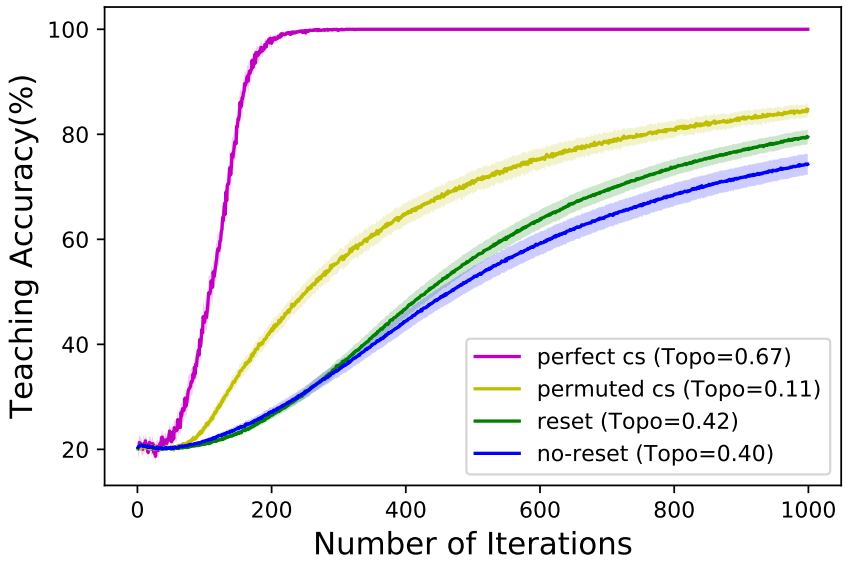

Figure 6.9: Ease-of-teaching when $\lambda^S = 0.05$ and $\lambda^L = 0.05$.

same until 180k iterations, and then starts dropping.

The reasons why languages are getting less easier-to-teach may be that we need more exploration for the speaker so that each time the listener gets reset, the language could possibly vary to a simpler one. To have that degree of exploration and keep each trial away from a pooling equilibrium at the same time is hard to obtain in this binary vocabulary setting. With a larger exploration hyperparameter, $\lambda^S = 0.1$ and $\lambda^L = 0.05$, for the reset regime there are 4 out of 100 trials producing degenerate languages, 2 of 4 trials jump out of this local optimum eventually. For the no-reset regime, 7 out of 100 trials stay in the local optimum of a pooling equilibrium eventually due to the bad initialization and the lack of additional force to explore the language. If we intentionally removed the runs that produce degenerate languages, and average the remaining trials for both regimes. We draw such post-processed teaching speed of the languages and the changes of topographic similarity in Figure 6.13 and Figure 6.12. The topographic similarity and the teaching speed both rise up for the reset regime, while drop for the no-reset regime. This is similar to what we have shown in previous chapters.

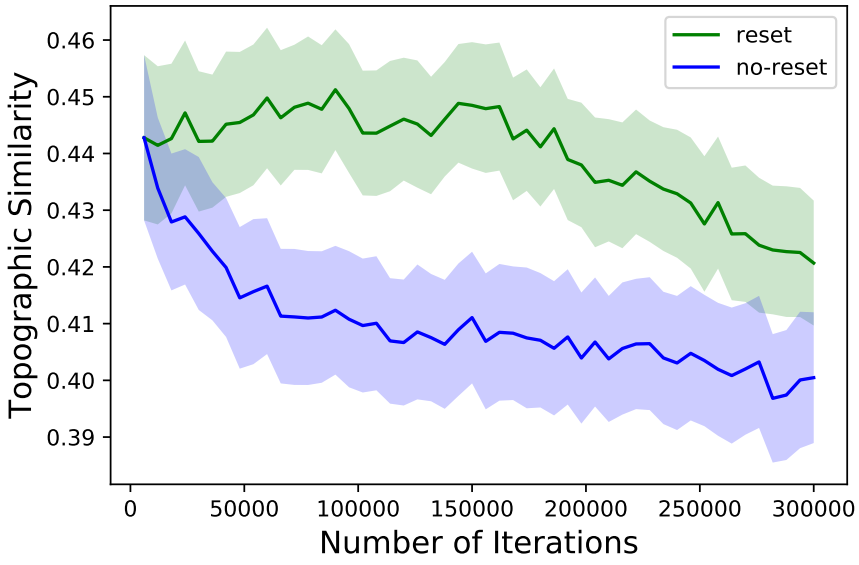

Figure 6.10: Topographic similarity when $\lambda^S = 0.05$ and $\lambda^L = 0.05$.

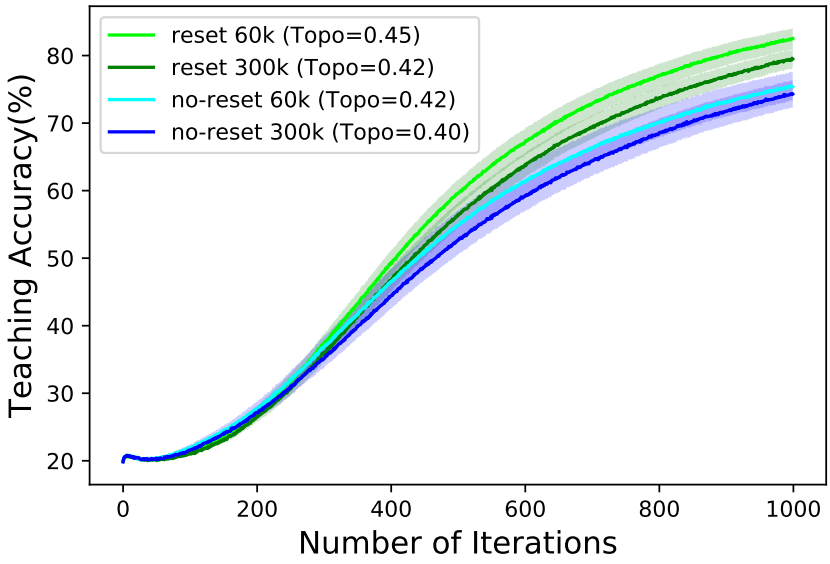

Figure 6.11: Ease-of-teaching during training when $\lambda^S = 0.05$ and $\lambda^L = 0.05$.

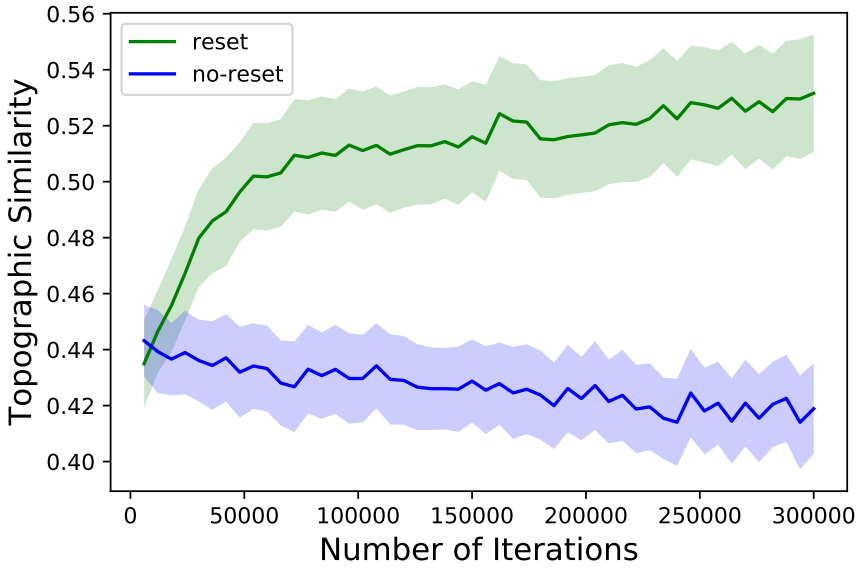

Figure 6.12: Topographic similarity when $\lambda^S = 0.1$ and $\lambda^L = 0.05$ (removed degenerate ones).

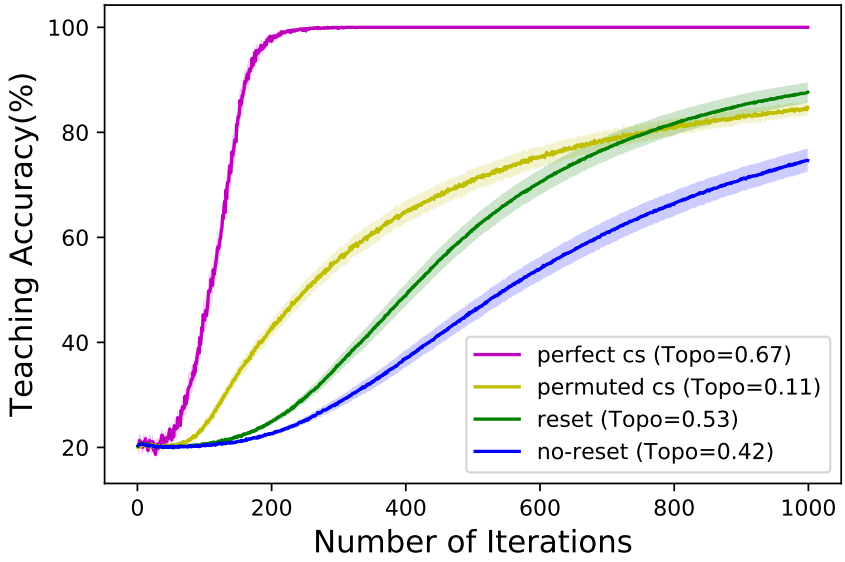

Figure 6.13: Ease-of-teaching when $\lambda^S = 0.1$ and $\lambda^L = 0.05$ (removed degenerate ones).

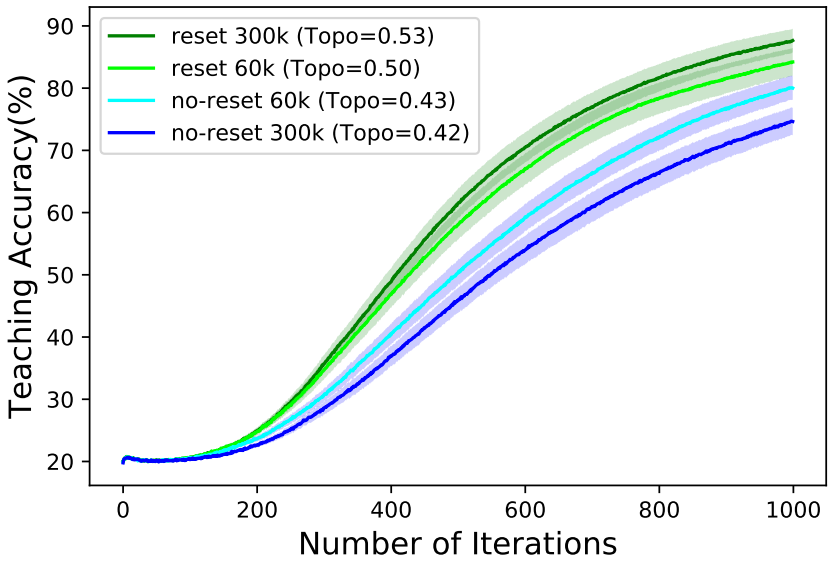

Figure 6.14: Ease-of-teaching during training when $\lambda^S = 0.1$ and $\lambda^L = 0.05$ (removed degenerate ones).

6.2.3 Can We Find "Words" in the Emergent Languages?

From the reset regime experiments when $\lambda^S = 0.1$ and $\lambda^L = 0.05$, 16 out of 100 trials converge to a perfect language with topographic similarity about 0.67 after training for 300k iterations. Table 6.1 shows an example of such a language. Every object is referred as different messages by the speaker in this language. Each color is expressed by the same word as the middle 3 bits. The first and the last bit are interpreted together as different shapes. Messages starting with '1' and ending with '0' mean 'circle', and starting with '0' and ending with '1' mean 'square', and starting with '0' and ending with '0' mean 'star', and starting with '1' and ending with '1' mean 'triangle'. This suggests that consecutive or inconsecutive bits of 'characters' can form together to represent an integral meaning and function similar to "words". Note that we do not find such resulting languages from the no-reset regime when $\lambda^S = 0.1$ and $\lambda^L = 0.05$, nor from any regimes when $\lambda^S = 0.05$ and $\lambda^L = 0.05$.

Table 6.1: A learned language with topographic similarity 0.67 in a limited vocabulary

	black	blue	green	grey	pink	purple	red	yellow
circle	10110	11110	11100	11010	10000	10010	11000	10100
square	00111	01111	01101	01011	00001	00011	01001	00101
star	00110	01110	01100	01010	00000	00010	01000	00100
triangle	10111	11111	11101	11011	10001	10011	11001	10101

6.2.4 Discussion

In the binary vocabulary setting, we find that there exists a dilemma between having large enough exploration for the speaker to vary languages and keeping stability of not reaching a pooling equilibrium. With different entropy hyperparameters, we get quite different results, which illustrates entropy hyperparameters affect the degree of impact. In this case, larger exploration combined with incorporating new listeners can find compositional languages. However, the drawback is large exploration may lead to a bad local minimum occasionally.

Compositionality can be seen as rising from a trade-off between pressures for compressibility and pressures for expressivity [31]. The appropriate pressure for compositionality is to see if compositional languages are dominant in the distribution of languages (including degenerate languages, holistic languages, compositional languages, and others). Degenerate languages are the most compressible ones. Holistic languages are the most expressive ones, but not compositional at all. Ease-of-teaching naturally considers both compressibility and expressivity, which builds a connection with compositionality directly.

To get a reliable emergence of optimal signaling (i.e., unambiguous language), a systemic bias against ambiguity is required after examining different models used in game theory, artificial life, evolutionary linguistics [55]. We do not encode such bias into our learning process, since it is out of the scope of this thesis. It may be helpful to avoid getting into the bad local minimum.

Chapter 7

Conclusions and Future Works

We propose new training regimes for the family of referential games to shape the emergent languages to be easier-to-teach and more compositional. We first introduce ease-of-teaching as a factor to evaluate emergent languages. We then show the connection between ease-of-teaching and compositionality, and how introducing new listeners periodically can create a pressure to increase both. We further experiment with resetting a single listener and all listeners within a population, and find that it is critical that new listeners are introduced abruptly rather than smoothly for the effect to be pronounced.

As future work to the emergence of compositional languages in referential games, a generalization test on held-out compositions of attributes can be conducted. And to develop a compositional language consistently, a bias against ambiguity should be considered. One idea is to incorporate curriculum learning, gradually increasing the difficulty of the referential tasks. We can increase the number of distractors or present distractors that are harder to distinguish to the agents.

Ease-of-teaching is a new metric to compare the communication protocols. We implicitly optimize ease-of-teaching in this thesis. Explicitly optimizing this measure should be considered, for example, using meta-learning to optimize the accuracy of teaching a new listener after a few updates. The training regime can be incorporated to develop compositional communication protocols, which can be more easily understood by humans. The connection between ease-of-teaching and compositionality should be explored when images as in-

put instead of symbolic input. Abrupt changes in the objective are somewhat helpful, and should be examined to what extent in a broader context.

This is the first work to consider the effect of introducing new agents in communication between RL agents. We explore it in the context of referential games. It would be interesting to examine if the effect of introducing new agents holds when behaviour and communication actions coexist in the environment. It may be used to produce simpler behaviour strategies other than communication for multiple agents as well.

References

[1] J. Andreas and D. Klein, "Analogs of linguistic structure in deep representations," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 2893–2897. DOI: 10.18653/v1/D17-1311. 1

[2] N. Bard, J. N. Foerster, S. Chandar, N. Burch, M. Lanctot, H. F. Song, E. Parisotto, V. Dumoulin, S. Moitra, E. Hughes, *et al.*, "The Hanabi challenge: A new frontier for AI research," *arXiv preprint arXiv:1902.00506*, 2019. 7

[3] R. Bellman, R. Bellman, and R. Corporation, *Dynamic Programming*, ser. Rand Corporation research study. Princeton University Press, 1957. 5

[4] B. Bogin, M. Geva, and J. Berant, "Emergence of communication in an interactive world with consistent speakers," *arXiv preprint arXiv:1809.00549*, 2018. 13

[5] H. Brighton and S. Kirby, "Understanding linguistic evolution by visualizing the emergence of topographic mappings," *Artificial life*, vol. 12, no. 2, pp. 229–242, 2006. 21

[6] K. Cao, A. Lazaridou, M. Lanctot, J. Z. Leibo, K. Tuyls, and S. Clark, "Emergent communication through negotiation," in *International Conference on Learning Representations*, 2018. 7

[7] E. Choi, A. Lazaridou, and N. de Freitas, "Multi-agent compositional communication learning from raw visual input," in *International Conference on Learning Representations*, 2018. 1, 8, 12, 13

[8] M. Cogswell, J. Lu, S. Lee, D. Parikh, and D. Batra, "Emergence of compositional language with deep generational transmission," *arXiv preprint arXiv:1904.09067*, 2019. 16

[9] V. P. Crawford and J. Sobel, "Strategic information transmission," *Econometrica: Journal of the Econometric Society*, pp. 1431–1451, 1982. 11

[10] A. Das, S. Kottur, J. M. Moura, S. Lee, and D. Batra, "Learning cooperative visual dialog agents with deep reinforcement learning," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2951–2960. 13

[11] A. Dosovitskiy and V. Koltun, "Learning to act by predicting the future," in *International Conference on Learning Representations*, 2017. 10

[12] K. Evtimova, A. Drozdov, D. Kiela, and K. Cho, "Emergent communication in a multi-modal, multi-step referential game," in *International Conference on Learning Representations*, 2018. 1, 13–15

[13] J. Foerster, I. A. Assael, N. de Freitas, and S. Whiteson, "Learning to communicate with deep multi-agent reinforcement learning," in *Advances in Neural Information Processing Systems*, 2016, pp. 2137–2145. 1, 8

[14] G. Frege, "Sense and reference," *The philosophical review*, vol. 57, no. 3, pp. 209–230, 1948. 1, 13

[15] C. L. Giles and K.-C. Jim, "Learning communication for multi-agent systems," in *Workshop on Radical Agent Concepts*, Springer, 2002, pp. 377–390. 8

[16] C. Guestrin, D. Koller, and R. Parr, "Multiagent planning with factored MDPs," in *Advances in Neural Information Processing Systems*, 2002, pp. 1523–1530. 8

[17] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," in *Advances in Neural Information Processing Systems*, 2017, pp. 5767–5777. 10

[18] M. Hausknecht and P. Stone, "Deep recurrent Q-learning for partially observable MDPs," in *2015 AAAI Fall Symposium Series*, 2015. 9

[19] S. Havrylov and I. Titov, "Emergence of language with multi-agent games: Learning to communicate with sequences of symbols," in *Advances in Neural Information Processing Systems*, 2017, pp. 2149–2159. 1, 12–14, 19

[20] K. M. Hermann, F. Hill, S. Green, F. Wang, R. Faulkner, H. Soyer, D. Szepesvari, W. M. Czarnecki, M. Jaderberg, D. Teplyashin, *et al.*, "Grounded language learning in a simulated 3D world," *arXiv preprint arXiv:1706.06551*, 2017. 8

[21] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997. 19

[22] M. Jaderberg, W. M. Czarnecki, I. Dunning, L. Marris, G. Lever, A. G. Castañeda, C. Beattie, N. C. Rabinowitz, A. S. Morcos, A. Ruderman, N. Sonnerat, T. Green, L. Deason, J. Z. Leibo, D. Silver, D. Hassabis, K. Kavukcuoglu, and T. Graepel, "Human-level performance in 3D multiplayer games with population-based reinforcement learning," *Science*, vol. 364, no. 6443, pp. 859–865, 2019. 2

[23] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," in *International Conference on Learning Representations*, 2017. 12

[24] N. Jaques, A. Lazaridou, E. Hughes, C. Gulcehre, P. Ortega, D. Strouse, J. Z. Leibo, and N. De Freitas, "Social influence as intrinsic motivation for multi-agent deep reinforcement learning," in *Proceedings of the 36th International Conference on Machine Learning*, 2019, pp. 3040–3049. 1, 7, 10, 11

[25] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial intelligence*, vol. 101, no. 1-2, pp. 99–134, 1998. 7

[26] T. Kasai, H. Tenmoto, and A. Kamiya, "Learning of communication codes in multi-agent reinforcement learning problem," in *2008 IEEE Conference on Soft Computing in Industrial Applications*, IEEE, 2008, pp. 1–6. 8

[27] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *International Conference on Learning Representations*, 2015. 7, 20

[28] S. Kirby, "Learning, bottlenecks and the evolution of recursive syntax," 2002. 16

[29] S. Kirby, H. Cornish, and K. Smith, "Cumulative cultural evolution in the laboratory: An experimental approach to the origins of structure in human language," *Proceedings of the National Academy of Sciences*, 2008. 16, 21, 43

[30] S. Kirby, T. Griffiths, and K. Smith, "Iterated learning and the evolution of language," *Current opinion in neurobiology*, vol. 28, pp. 108–114, 2014. 16

[31] S. Kirby, M. Tamariz, H. Cornish, and K. Smith, "Compression and communication in the cultural evolution of linguistic structure," *Cognition*, vol. 141, pp. 87–102, 2015. 1, 2, 16, 42, 48

[32] V. R. Konda and J. N. Tsitsiklis, "Actor-critic algorithms," in *Advances in neural information processing systems*, 2000, pp. 1008–1014. 7

[33] S. Kottur, J. Moura, S. Lee, and D. Batra, "Natural language does not emerge 'naturally' in multi-agent dialog," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 2962–2967. DOI: 10.18653/v1/D17-1321. 1, 8, 12–14, 18

[34] A. Lazaridou, K. M. Hermann, K. Tuyls, and S. Clark, "Emergence of linguistic communication from referential games with symbolic and pixel input," in *International Conference on Learning Representations*, 2018. 1, 8, 12–15, 19, 21

[35] A. Lazaridou, A. Peysakhovich, and M. Baroni, "Multi-agent cooperation and the emergence of (natural) language," *International Conference on Learning Representations*, 2017. 1, 13

[36] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015. 4

[37] J. Lee, K. Cho, J. Weston, and D. Kiela, "Emergent translation in multi-agent communication," in *International Conference on Learning Representations*, 2018. 1, 13

[38] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-end training of deep visuomotor policies," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 1334–1373, 2016. 4

[39] D. Lewis, *Convention: A philosophical study*. John Wiley & Sons, 2008. 11

[40] R. Lowe, J. Foerster, Y.-L. Boureau, J. Pineau, and Y. Dauphin, "On the pitfalls of measuring emergent communication," in *Proceedings of the 18th International Conference on Autonomous Agents and Multi-Agent Systems*, International Foundation for Autonomous Agents and Multiagent Systems, 2019, pp. 693–701. 11

[41] R. Lowe, Y. Wu, A. Tamar, J. Harb, O. P. Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," in *Advances in Neural Information Processing Systems*, 2017, pp. 6379–6390. 7

[42] C. J. Maddison, A. Mnih, and Y. W. Teh, "The concrete distribution: A continuous relaxation of discrete random variables," in *International Conference on Learning Representations*, 2017. 10

[43] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *International Conference on Machine Learning*, 2016, pp. 1928–1937. 20

[44] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015. 4, 9, 29

[45] I. Mordatch and P. Abbeel, "Emergence of grounded compositional language in multi-agent populations," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 1, 8, 10, 12–15

[46] M. A. Nowak, J. B. Plotkin, and V. A. Jansen, "The evolution of syntactic communication," *Nature*, vol. 404, no. 6777, p. 495, 2000. 14

[47] C. Pawlowitsch, "Why evolution does not always lead to an optimal signaling system," *Games and Economic Behavior*, vol. 63, no. 1, pp. 203–226, 2008. 11

[48] T. Sasaki and D. Biro, "Cumulative culture can emerge from collective intelligence in animal groups," *Nature communications*, vol. 8, p. 15 049, 2017. 16

[49] D. Silver, H. van Hasselt, M. Hessel, T. Schaul, A. Guez, T. Harley, G. Dulac-Arnold, D. Reichert, N. Rabinowitz, A. Barreto, *et al.*, "The predictron: End-to-end learning and planning," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 2017, pp. 3191–3199. 10

[50] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, Jan. 2016. DOI: 10.1038/nature16961. 4

[51] J. Sirota, "Evolving recurrent neural networks for emergent communication," 2019. 13

[52] B. Skyrms, *Signals: Evolution, learning, and information*. Oxford University Press, 2010. 12

[53] K. Smith, H. Brighton, and S. Kirby, "Complex systems in language evolution: The cultural emergence of compositional structure," *Advances in Complex Systems*, vol. 6, no. 04, pp. 537–558, 2003. 1

[54] K. Smith, S. Kirby, and H. Brighton, "Iterated learning: A framework for the emergence of language," *Artificial life*, vol. 9, no. 4, pp. 371–386, 2003. 16

[55] M. Spike, K. Stadler, S. Kirby, and K. Smith, "Minimal requirements for the emergence of learned signaling," *Cognitive science*, vol. 41, no. 3, pp. 623–658, 2017. 48

[56] S. Sukhbaatar, R. Fergus, *et al.*, "Learning multiagent communication with backpropagation," in *Advances in Neural Information Processing Systems*, 2016, pp. 2244–2252. 1, 8, 9

[57] R. S. Sutton, "Learning to predict by the methods of temporal differences," *Machine Learning*, vol. 3, no. 1, pp. 9–44, 1988, ISSN: 1573-0565. DOI: 10.1007/BF00115009. 5

[58] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. 2018. 4, 7

[59] R. S. Sutton, D. A. McAllester, S. P. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," in *Advances in Neural Information Processing Systems*, 2000, pp. 1057–1063. 6

[60] P. Varshavskaya, L. P. Kaelbling, and D. Rus, "Efficient distributed reinforcement learning through agreement," in *Distributed Autonomous Robotic Systems 8*, Springer, 2009, pp. 367–378. 8

[61] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine learning*, vol. 8, no. 3-4, pp. 229–256, 1992. 6, 20