Trading off Utility, Informativeness, and Complexity in Emergent Communication

Anonymous Author(s)
Affiliation
Address
email

Abstract

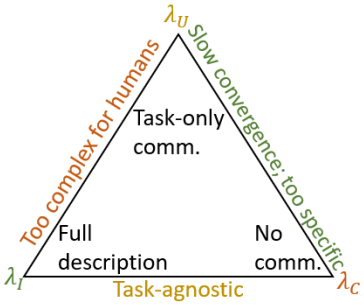
Emergent communication research often focuses on optimizing task-specific utility as a driver for communication. However, there is substantial evidence that human languages evolve under pressure to efficiently compress meanings into communication signals by optimizing the Information Bottleneck tradeoff between informativeness and complexity. In this work, we study how trading off these three factors — utility, informativeness, and complexity — shapes emergent communication, including compared to human communication. To this end, we propose Vector-Quantized Variational Information Bottleneck (VQ-VIB), a method for training neural agents to compress inputs into discrete signals embedded in a continuous space. We train agents via VQ-VIB and compare their performance to previously proposed neural architectures, both in multi-agent reinforcement learning settings and in a Lewis reference game. Across all neural architectures and settings, taking into account communicative informativeness benefits communication convergence rates, and penalizing communicative complexity leads to human-like lexicon sizes while maintaining high utility. Additionally, we find that VQ-VIB outperforms other discrete communication methods. This work demonstrates how fundamental principles that are believed to characterize human language evolution may inform emergent communication in artificial agents.

1 Introduction

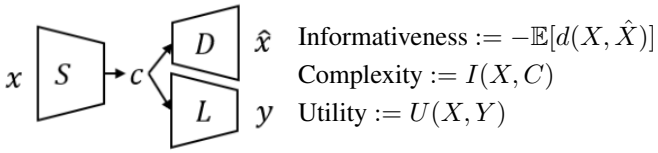
Good communication is a critical component of successful teams of humans and artificial agents, but differing notions of "good" makes training agents to develop such communication challenging. One view of communication focuses on *utility* by framing languages as successful to the extent they enable high task performance. Emergent communication literature, wherein agents learn to communicate while optimizing a task-specific reward or utility, emphasizes this view [1–6]. However, training via utility alone may lead to slow convergence rates [7], and aspects of learned communication are highly sensitive to environment design (at times diverging from human naming patterns [8]) implying that additional constraints are needed in order to guide agents toward human-like communication. Finding such constraints is an open question, both for artificial intelligence and for cognitive modeling.

Here, we address this question by building on a recent line of work that argues that human languages evolve under pressure to efficiently compress meanings into communicative signals [9–13]. This notion of efficiency is formulated in terms of the Information Bottleneck (IB) principle [14], which is a general information-theoretic principle with broad scope in machine learning [15–19], and in this context, it can be interpreted as a tradeoff between the complexity and informativeness of

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



(a) Theoretical Framework



(b) Simplified Architecture. A speaker, S, maps an input, x, to a communication signal, c. A listener, L, takes an action, y, and a decoder, D, reconstructs x. Informativeness is negatively related to the expected distortion between x and $\hat{x}$; complexity is defined as the mutual information between X and C; utility can be any given task-specific reward function or loss.

Figure 1: Our theoretical framework and training losses allow us to weigh tradeoffs between utility (λ_U), informativeness (λ_I), and complexity (λ_C) to guide properties of emergent communication.

communication [9]. Intuitively, complexity corresponds to the number of bits that are needed for communication, and informativeness corresponds to how well a listener can infer the speaker's intended meaning regardless of a specific task (e.g., humans understand the term "blue" in "avoid the blue box" and "find a blue cup"). We hypothesize that taking into account complexity and informativeness, in addition to optimizing a task-specific utility function, may help guide artificial agents toward more natural, human-like semantic systems (e.g., learning a single word for "blue" rather than 20 unique words for very similar shades of blue).

We test this idea by training teams of artificial agents to optimize a tradeoff between maximizing task utility, maximizing communicative informativeness, and minimizing communicative complexity in emergent communication settings, as depicted in Figure 1. To this end, we extend prior work [20, 21] and propose Vector-Quantized Variational Information Bottleneck (VQ-VIB), a method for training neural agents to compress inputs into signals, while maintaining a discrete (symbolic) signal representation that is embedded in a continuous space. In experiments, we find that training to increase informativeness improves the convergence rate of emergent communication to high-performance communication, while annealing the complexity loss generates a spectrum of learned communication, which we show aligns with a variety of human languages. Additionally, we find that VQ-VIB appears better able to achieve high utility and informativeness for the same complexity than other discrete communication methods.

2 Related Work

We train neural agents, focusing on the effects of utility, informativeness, and complexity on emergent communication. Here, we describe related work in emergent communication; we include approaches for measuring informativeness and complexity in Section 3. Generally, in emergent communication settings, researchers train agents to accomplish a cooperative task in which communication should enable better task performance [3, 22, 23]. For example, if one agent observes a goal location that another agent must navigate to, the team of agents would benefit from the "speaker" agent communicating with the "listener" agent. Many emergent communication works have considered how environmental factors (such as reward, agent architecture, or more) affect learned communication [3, 5, 22, 24]. We focus on subsets of the literature that explore aspects of informativeness and complexity, as well as specific applications in color reference games.

Notions of Complexity in Emergent Communication. Recent emergent communication literature has taken initial steps beyond focusing only on utility by limiting complexity or encouraging informativeness, although these terms are rarely explicitly acknowledged. There are many notions of a language's complexity, including aspects of grammar, syntax, morphology, length, or compositionality [25–28]; we do not position ourselves within literature that seeks to define complexity. Rather, we adopt a specific, existing metric of complexity, which we define more fully in Section 3. Numerous

works simplify communication by discretizing it, adding noise to the environment, or penalizing large vocabulary sizes; these methods all align with limiting complexity [21, 29, 30]. Wang et al. [31] explicitly limit complexity via a centralized message processor that acts as an information bottleneck. At the same time, other works find that auxiliary losses designed to create "meaningful" communication improve convergence rates [7, 32]. For example, Lin et al. [33] use an autoencoding loss to reconstruct states from communication; this is tightly related to notions of informativeness. However, because Lin et al. [33] do not penalize communication complexity, they note that their method "may fail... where agents need to be more selective of the information communicated." Lastly, Chaabouni et al. [8] measure (but do not directly control) emergent communication complexity while changing environment parameters to induce different forms of communication. Thus, while emergent communication literature has begun to rediscover the importance of complexity and informativeness in communication, these terms have rarely been explicitly recognized, especially in training.

Emergent Communication and Color Reference Games. We conduct experiments in color reference games, which have been widely used in emergent communication literature [1, 34, 35, 8]. Typically, these reference games, or Lewis signalling games [36], comprise a speaker, who observes one color dubbed the "target," and a listener, who observes the target color and distractor color(s). By communicating to the listener, the speaker seeks to have the listener identify the target. We expand upon our specific experiment configuration in Section 4.4.

Color naming has been a central test-case for cognitive theories of semantic categorization and language evolution [37]. It is a rare domain in which a large-scale cross-linguistic dataset, the World Color Survey [WCS; 38], is available, documenting the wide diversity in color naming schemes across languages. Particularly relevant to our work is Zaslavsky et al. [9]'s finding that color naming systems across languages attain complexity–informativeness tradeoffs that are near-optimal in the Information Bottleneck [IB; 14] sense. Building on that, Chaabouni et al. [8] showed that AI agents can learn IB-efficient color naming systems by playing color reference games (and see [35] for a related study using different notions of efficiency). Unlike Chaabouni et al. [8], who studied the effect of environment design on communicative informativeness and complexity, we directly train agents via a tradeoff of these terms without changing the environment. In addition, we propose a new agent architecture that achieves higher utility for a given complexity than other methods.

3 Technical Approach

Here, we show how to adapt standard neural architectures to enable informativeness- and complexity-based objectives. We then introduce a new neural architecture, Vector-Quantized Variational Information Bottleneck (VQ-VIB), for learning complexity-limited discrete representations in a continuous space. Lastly, we combine utility, informativeness, and complexity terms into an overall objective.

3.1 The Information Bottleneck Framework for Semantic Systems

We adopt and extend Zaslavsky et al. [9]'s IB framework for semantic systems, which has substantial empirical support across languages and domains [10–13]. In this framework, speakers and listeners jointly optimize the IB tradeoff between the complexity and informativeness of their communication system. A speaker is characterized as a probabilistic encoder $q(c|x)$ that, given an input, $x \sim p(x)$, generates a signal, c. A listener is characterized as a probabilistic decoder $q(\hat{x}|c)$, that given c aims to reconstruct the speaker's input. Complexity is measured by the mutual information between the speaker's inputs and signals, $I(X, C)$. Informativeness is measured by the degree to which the speaker's mental representation of x matches the listener's mental representation of $\hat{x}$. In general, these mental representations are defined by probability distributions, m and $\hat{m}$ respectively, over some feature space. In that case, minimizing the expected Kullback-Leibler (KL) divergence $\mathbb{E}[D[m\|\hat{m}]]$ amounts to maximizing the informativeness of the communication. This yields the following optimization principle: $\min I(X, C) - \beta\mathbb{E}[D[m\|\hat{m}]]$, where β is a tradeoff parameter. It has been shown that this principle is equivalent to the IB principle [9].

Solving this optimization problem directly is challenging in large scale domains and with agents that are parameterized by neural networks. To address this, we take a Variational Information bottleneck [VIB; 18] approach, in which an upper bound is optimized, similar to ELBO in Variational Autoencoders [VAEs, 39] and β-VAEs [40]. In practice, we also approximate informativeness by an autoencoding loss, $d(x, \hat{x})$, defined by the mean squared error (MSE) loss of the speaker's input and the listener's reconstruction. We next explain our approach in more detail.

3.2 Vector Quantized Variational Information Bottleneck (VQ-VIB)

Our proposed neural architecture, the Vector Quantized Variational Information Bottleneck (VQ-VIB), learns information-constrained discrete communication in a continuous space. Our technique combines prior work in discrete representation learning [VQ-VAE, 20] with VIB [18].

3.2.1 Vector-Quantized Variational Autoencoder

First, we summarize a simplified Vector-Quantized Variational Autoencoder (VQ-VAE) architecture [20]. As in a standard VAE, our encoder is a conditional distribution over latent representations, given an input: $q(z|x)$. A deterministic feedforward network maps from an input, x, to a latent representation, z: we dub such encodings, $z(x)$. Next, for a VQ-VAE parametrized with K embedding vectors, $\boldsymbol{\zeta}$, in the latent space, $z(x)$ is discretized to a particular embedding vector, ζ_i, by mapping to the nearest embedding vector. Thus, the discretized embedding is determined by Equation 1.

$$q(z = \zeta_i|x) = \begin{cases} 1 & \text{if} \quad \zeta_i = \text{argmin}_{\zeta_j}[(z(x) - \zeta_j)^2] \\ 0 & \text{else} \end{cases} \tag{1}$$

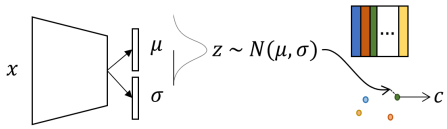
3.2.2 Variational Information Bottleneck (VIB)

We also use notions from VIB, a variational approach for generating information-constrained continuous representations with neural nets. Following Alemi et al. [18], an encoder network maps from an input to μ and σ, which are used as parameters to a Gaussian distribution from which a latent representation, z, is sampled. VIB networks are trained via a weighted combination of a classification loss and (β times) the KL divergence of $\mu(x)$ and $\sigma(x)$ from a unit Gaussian; this KL loss corresponds to an upper bound of $I(X, Z)$.

3.2.3 VQ-VIB Architecture

Here, we integrate VQ-VAE and VIB for a novel method of generating discrete, information-bounded representations. In the **VQ-VIB** method, an encoder is used to sample a representation z in a continuous space, followed by a quantization layer that discretizes the sample by mapping to the nearest encoding ζ. This architecture is depicted in Figure 2 and is formally defined in Equation 2.

In the VQ-VIB formulation, the likelihood of a communication vector, c, given an input, x, corresponds to the likelihood that an encoding z, sampled with parameters $\mu(x)$ and $\sigma(x)$, is closest to an embedding vector equal to c. Intuitively, this approach corresponds to uncertainty over which "word" to use. One may compute an upper bound on $I(X, C)$ using the standard KL divergence used for Gaussians in a VIB, simply using $\mu(x)$ and $\sigma(x)$.



$$\begin{aligned} &z \sim \mathcal{N}(\mu(x), \sigma(x)) \\ &q_{SB}(c|x) = \mathbb{P}(c = \text{argmin}_{\zeta}[(z - \zeta)^2] \end{aligned} \tag{2}$$

Figure 2: The VQ-VIB architecture blends Vector Quantization and Variational Information Bottleneck. The speaker encodes an input, x, as parameters to a Gaussian, from which a vector is sampled and then discretized by mapping to the nearest encoding, to produce communication, c.

We proposed an additional network architecture, dubbed **VQ-VIB After**, that is related to our main VQ-VIB method but relates to different forms of bounded complexity and does not generate discrete communication vectors. We included discussion of this approach in Appendix B; in our main paper, we include results of using VQ-VIB After as a baseline, but again emphasize that the approach is not related to the core focus of our work on discrete emergent communication. To prevent ambiguity in presenting our results, at times we label VQ-VIB results as "VQ-VIB Before" to distinguish our standard method, in which sampling is performed before discretization, from VQ-VIB After.

3.2.4 Complexity Equivalence Classes

Although one may approximate $I(X, C)$, simply penalizing this term in training was insufficient to train VQ-VIB agents to use fewer unique discrete embeddings. Fundamentally, two agents could belong to the same complexity equivalence class (with the same $I(X, C)$) but have different numbers of embeddings if several embeddings are nearly identical (see [41] for analysis of these equivalence classes in IB). Intuitively, this corresponds to notions of either communicating with many words, each of which is difficult to disambiguate from other words, or fewer words, each of which is quite distinct. We wish to bias agents to use fewer distinct clusters, which corresponds to the entropy over the categorical distribution over embeddings. Training with the exact entropy in the VQ-VIB architecture is infeasible given the non-differentiablity of the `argmin` operation used in discretization. Instead, we use the entropy of an approximation of the distribution over embeddings: the softmax of the negative L2 distance from embeddings to each token. (We leave exploration of other approximations to future work.) Because this loss is only used to change the number of clusters within a complexity equivalence class, we require only a small training signal. In our experiments, we multiply this loss by 0.05, making it orders of magnitude smaller than other losses.

3.3 Overall Optimization

Here, we combine losses into an idealized objective function to maximize, shown in Equation 3. This objective function is the weighted combination of three terms, wherein we seek to maximize a tradeoff between high utility, high informativeness (low expected distortion d), and low complexity. Intuitively, these three scalar weights form a simplex that describe a range of communication from silence (minimizing complexity) to task-optimized communication that does not generalize (maximizing utility) to complete descriptions (maximizing informativeness).

$$\text{maximize} \quad \lambda_U U(X, Y) - \lambda_I \mathbb{E}[d(X, \hat{X})] - \lambda_C I(X, C) \tag{3}$$

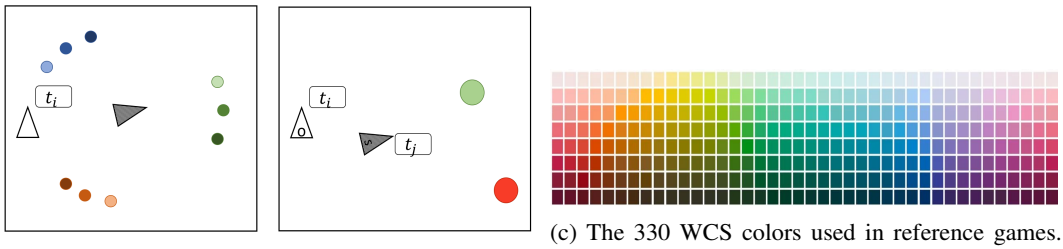
In practice, we train neural agents via gradient descent to maximize differentiable approximations of these terms. Utility is represented via policy gradient methods (for MARL settings) or crossentropy loss. Informativeness is computed via the negative mean squared error between the speaker's input and a decoder's output, given the communication. Lastly, an upper bound of complexity is computed as the KL divergence between the speaker's distribution over communication and a fixed marginal (either a uniform distribution for onehot communication, or a unit Gaussian for others).

4 Experiment Preliminaries

We formulated hypotheses about the roles of utility, informativeness, and complexity in emergent communication, and designed experiments to test them.

4.1 Hypotheses

We expected that increasing informativeness is a bias for learning useful communication, but that doing so would likely increase complexity, which must be bounded to align with human communication. More specifically, our two hypotheses were:



(a) MARL: 9 Points (b) MARL: Uniform (c) The 330 WCS colors used in reference games. Agents observed a color's 3D CIELAB coordinates.

Figure 3: The three test environments. In the MARL experiments (a and b) a speaker agent (clear) communicated a target location to a listener (gray) (figures taken from Tucker et al. [21]). In the color environment, agents observed colors from the WCS Dataset [38].

- Training to increase informativeness will cause agents to converge to successful communication systems more quickly than without such training, although increasing informativeness will increase complexity.
- Complexity-bounded agents will generate more human-like discrete communication systems than unbounded agents.

4.2 Trials and Baselines

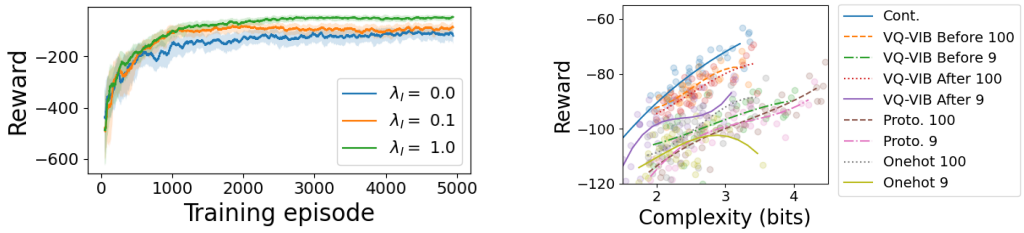
We compared 5 types of neural architectures in our experiments: continuous, onehot, Proto (based on the prototype-based communication from Tucker et al. [21]), VQ-VIB, and VQ-VIB After. We again emphasize that only VQ-VIB and onehot communicate via a finite set of symbols, whereas other methods fall out of the scope of this work of complexity-bounded, *discrete* emergent communication. For each domain, we trained agents from scratch using 5 random seeds, starting at 0. As a high-level overview, Proto. agents internally compute a onehot vector and then multiply it by a learned "embedding" matrix, which generates a finite set of vectors in continuous space. Full details of the architectures, hyperparameters, and environments are included in Appendix A.

4.3 Particle Worlds

We trained agents in two environments inspired by those introduced by Tucker et al. [21]: 9 points and Uniform. Both environments consisted of two-dimensional worlds in which a speaker observed a target location that a listener had to reach but could not directly observe, as depicted in Figure 3. In the 9 points environment, there were 9 landmarks at fixed locations, one of which was chosen at random as the target for each episode. In the Uniform environment, two landmarks were spawned in locations drawn uniformly at random in the world; one landmark was specified as the target. Agents received identical reward equal to the negative distance from the listener to the target and were trained using Multi-Agent Deep Deterministic Policy Gradient [3]. The listener could not observe the target location, and the speaker could only communicate at the first timestep by broadcasting a vector that was observable at all future timesteps; thus, an optimal team policy would consist of the speaker communicating about the target at the first timestep, and the listener navigating straight to the target based on the message. The two environments differ in their complexity: in the 9 points environment, the speaker had a finite amount of information about which to communicate, whereas in the Uniform environment, the location could be any continuous value (up to the computer's float precision).

4.4 Color Reference Games

We also tested agents in reference games, or Lewis signalling games [36], which are often used in emergent communication studies; by focusing on a color domain, which has corresponding data on human communication [37, 38], we hoped to identify similarities between agent and human languages [35, 8]. Human languages use a finite set of words, so we only tested onehot and VQ-VIB



(a) Learning curves for VQ-VIB

(b) Complexity-Utility tradeoffs

Figure 4: Uniform environment results. a) Increasing λ_I improved convergence rates and utility (reward) during training (shown here for VQ-VIB). b) For fixed $\lambda_I = 1.0$, annealing λ_C generated a spectrum of agents from high-complexity, high-reward to low-complexity, low-reward.

communication. Other architectures (continuous, Proto, and VQ-VIB After) sample from continuous spaces and could therefore generate infinite unique communication vectors.

In these reference games, we trained agents within a supervised learning framework, differentiating through both the speaker and listener. A diagram of the network architecture is included in Figure 7 in Appendix A. A speaker observed a target color and (if in the "Common Ground" setting, explained later) a distractor color and output a communication vector. A listener observed the speaker's communication and the target and distractor colors and predicted which of the colors was the target. Colors were represented by their 3D CIELAB coordinates (illustrated in Figure 7 in Appendix A), and speaker observations were corrupted by zero-mean Gaussian noise with a standard deviation of 8, as motivated by human perception [9]. Both colors were sampled from the set of 330 World Color Survey (WCS) colors according to a human-based prior from [9].[1] Only the speaker agent observed which color was the target, so the agents had to learn to convey information from the speaker to the listener. Agents were trained via binary crossentropy loss computed on the listener's output, which worked more reliably than MARL methods (corroborating Chaabouni et al. [8]'s findings).

We trained agents in two settings: Isolated, in which the speaker only observed the target, and Common Ground, in which the speaker observed both the target and distractor colors. The Isolated scenario resembles the WCS setting, in which participants were asked to name individual colors. Conversely, the Common Ground scenario was inspired by work in pragmatics and Rational Speech Act, wherein humans communicate less information than fully describes a scene, but enough to induce listeners to behave in the desired manner [42, 43]. We note that, unlike Chaabouni et al. [8], we do not change the subset of sampled distractor colors to induce different complexity; our framework results in more stable training and allows us to directly vary communication complexity within the same environment.

5 Experiment Results

Our results indicated that an informativeness loss improved convergence, annealing λ_C generated a suite of agents of different complexities that more closely approximated human languages, and VQ-VIB agents achieved greater utility, for the same complexity, than onehot agents. [2]

5.1 Particle Worlds

In the particle worlds, increasingly penalizing complexity led agents to lower-utility communication, although architecture choices and training to increase informativeness could increase reward. First, we conducted experiments in which we did not penalize complexity, but varied λ_I across trials.

[1] The WCS data are available at `www1.icsi.berkeley.edu/wcs` and the human-based color prior is available at `github.com/nogazs/ib-color-naming`.

[2] Anonymized code is available at https://anonymous.4open.science/r/vqvib-NeurIPS-0F69

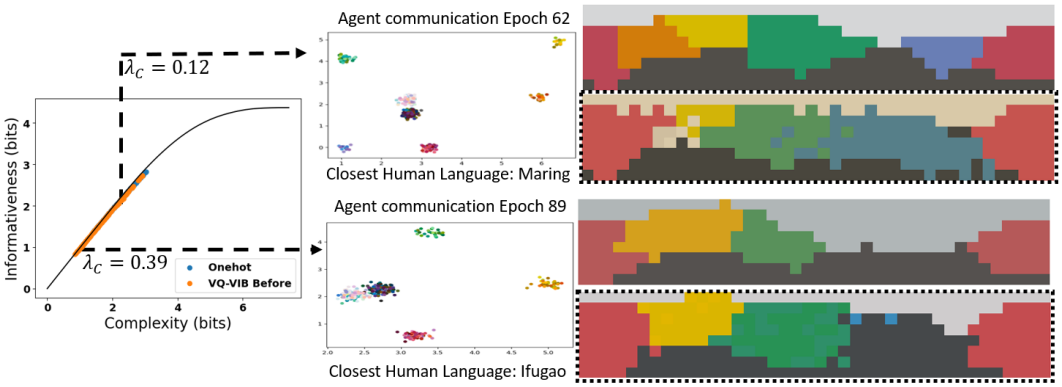


Figure 5: Isolated color reference game results. Left: By annealing complexity, we recreated a range of near-optimal color-naming schemes (black curve is the theoretical IB bound from [9]). Middle: 2D PCA of VQ-VIB communication shows relationships between different clusters in the communication space. Right: mode maps of the agent (no box) and human (dashed box) color naming systems, where each color chip from Figure 3c is colored by the centroid of its modal category.

Across all architectures, higher λ_I increased the mean team reward and improved convergence rates, as shown in the learning curves of VQ-VIB in Figure 4 a. We note that successfully training agents with a mix of informativeness and utility loss represents an improvement over Lin et al. [33], who struggled to train agents with a combination of losses.

Next, for a fixed $\lambda_I = 1.0$ (to improve convergence), we trained agents while annealing λ_C; this smoothly decreased the complexity of communication. Results for the Uniform environment aligned with our expectations Figure 4 b. Each curve indicates the cubic best fit to the data generated across 5 random seeds for each configuration (discarding outliers for reward less than -120), with complexity and reward measured at each increment of 1,000 training episodes from 3,000 to 10,000. As complexity decreased (moving right to left in the plot), reward decreased as well. Visualizations of agent behavior confirmed that low-complexity VQ-VIB agents communicated via increasingly coarse discretizations of the space and are included in Appendix C. The plots align with notions of how humans might communicate in natural language about a continuous space via coarse discretizations like “north” or “south” rather than hundreds of specific words for any possible location.

Lastly, these results suggest that neural architecture choice was important. In both 9 points and Uniform, the VQ-VIB agents tended to achieve higher reward, for the same complexity, as onehot or Proto agents. (Continuous communication reflected an optimal upper bound but violated our desire to learn discrete communication.) Furthermore, adding more encodings (9 discrete encodings vs. 100) significantly increased VQ-VIB performance, while changing onehot agents to communicate in 9 or 100 dimensions had a smaller effect. Parametrizing VQ-VIB or onehot agents with even more discrete tokens (200 or 500) did not improve mean utility beyond 100 tokens, suggesting that greater parametrization will not decrease the gap. This section includes partial results, focusing on VQ-VIB in the Uniform environment; full results are in Appendix C. We observed similar benefits from high λ_I, similar complexity-reward tradeoffs, and similar advantages of VQ-VIB over other architectures in the 9 points environment.

5.2 Reference Games

In color domains, encouraging informativeness and penalizing complexity generated a range of communication that closely approximated optimal compression and human languages.

5.2.1 Isolated Color Reference Game

In the Isolated color reference game, an informativeness training loss appeared necessary for onehot, while a complexity-based loss was useful for generating a spectrum of communication conventions.

Table 1: Isolated reference game results. **Top:** Mean (std) gNID between human languages and nearest agent. Annealing generated diverse complexities, aligning with human languages (left two columns represent no annealing). **Bottom:** Mean ϵ (deviation from optimality) for agents. Annealed agents were closer to the bound than fixed agents (lower ϵ).

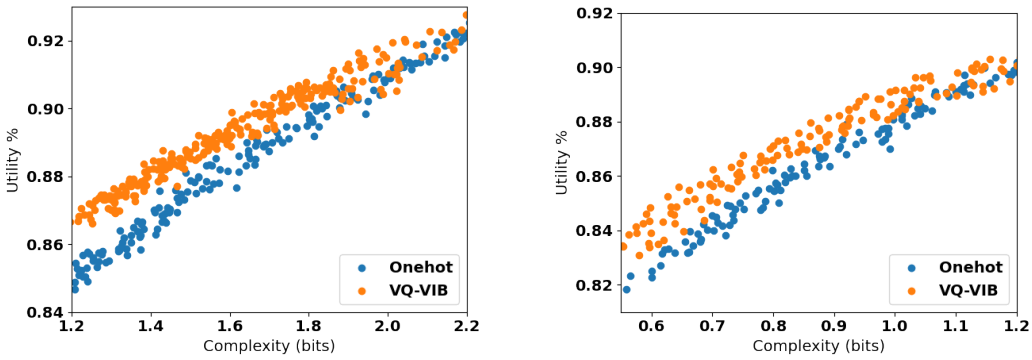
	VQ-VIB Fixed	Onehot Fixed	VQ-VIB Annealed	Onehot Annealed
Human gNID	0.29 (0.03)	0.38 (0.01)	**0.16** (0.01)	**0.16** (0.00)
Agent ϵ	0.08 (0.01)	0.08 (0.00)	**0.03** (0.00)	0.05 (0.00)

First, we trained agents with a range of λ_I values while leaving $\lambda_C = 0$. We briefly report two main findings: 1) for a given λ_I, agents converged to a small range of complexity and informativeness and 2) VQ-VIB agents typically learned more complex and informative communication than onehot agents did, although increasing λ_I shrank the gap. These experiments are similar to those conducted by Chaabouni et al. [8], who did not train with informativeness or complexity losses. Full results for these experiments are included in Appendix D.

Second, we tested agents for $\lambda_I = 1.0$ while linearly annealing λ_C to penalize complexity. As λ_C increased, the communication complexity decreased, as shown in Figure 5 (left). Over 5 random seeds, the points in the plot, generated at each training epoch, closely approximated the IB upper bound for maximum informativeness for a given complexity. (A zoomed-in version of similar analysis, conducted with 1024 possible tokens, is included in Figure 14.) Inspection of the learned color naming schemes at different complexity levels revealed important trends (Figure 5 right). As complexity decreased, the number of communication clusters emitted by VQ-VIB agents decreased (visualized via 2-dimensional principle component analysis). Clustering in the communication space was reflected in the color space (e.g., "yellow" and "orange" merging). More generally, a least-squares linear correlation between the distance between pairs of communication vectors and the distance between associated colors (in CIELAB space) showed $r^2 = 0.97 \pm (0.01)$ for VQ-VIB, whereas for onehot communication $r^2 = 0.76 \pm (0.03)$. That is, for VQ-VIB, if two communication vectors were far apart in communication space, they described colors that were far apart in CIELAB space. This shows a fundamental connection between the form and meaning of communication used by VQ-VIB agents, which onehot-based communication is architecturally unable to support.

Lastly, we compared agent color naming schemes to each other and to human language data. For each of the 110 languages in the WCS Dataset [38], we found the best-matching agent as determined by minimizing gNID, a measure of dissimilarity of naming distributions introduced by Zaslavsky et al. [9]. Example nearest languages are shown in Figure 5 in dashed boxes. Despite having no supervised data, the artificial agents learned similar color naming schemes to humans, but only when agents changed complexity to match human languages' (shown via the gNID metrics in Table 1). This highlights the importance of controlling complexity in emergent communication: only by changing complexity could we generate naming systems similar to human communication. Thus, methods like Lin et al. [33] that encourage informativeness without penalizing complexity would be unable to recreate our results, and Chaabouni et al. [8] would have to change the environment to indirectly control complexity. Finally, when comparing VQ-VIB and onehot naming schemes, we found that agents were similarly efficient (deviation from optimality is calculated as ϵ in Table 1), but that VQ-VIB achieved higher utility, for the same complexity, as shown in Figure 6. Varying the number of tokens from 330 to 1024 or replacing the supervised training mechanism with REINFORCE-based training did not change results, as examined in Appendix D. This corroborates patterns from the particle world environments, wherein VQ-VIB agents achieved higher rewward, for the same complexity, as other methods.

Together, these results indicate that annealing complexity is a critical component in recreating human-like color naming systems. Furthermore, VQ-VIB matched the form and meaning of communication by merging clusters in the communication space that represented nearby colors, and outperformed onehot communication by achieving higher utility for the same complexity.



(a) Isolated

(b) Common Ground

Figure 6: In both the Isolated (left) and Common Ground (right) color reference games, VQ-VIB achieved greater utility, for the same complexity, than onehot communication.

5.2.2 Common Ground Color Reference Game

We tested agents in the Common Ground color reference game to evaluate contextualized communication. Crucially, these experiments tested agent behavior when utility and informativeness were not identical: utility could be maximized by the speaker only communicating about the target color, whereas informativeness was maximized by communicating about both target and distractor. In this distinct setting, the same trends as in the Isolated experiments held. An informativeness loss improved convergence, and, while both VQ-VIB and onehot were nearly optimally efficient, VQ-VIB achieved greater utility than onehot did, as shown in Figure 6. Complete results and figures are included in Appendix D, confirming that agents learned efficient, contextualized communication.

6 Contributions

In this work, we explore the interplay of utility, informativeness, and complexity in emergent communication. To the best of our knowledge, we are the first to directly optimize agents to communicate according to a weighted combination of all three terms. In our experiments, we found that training agents while rewarding informativeness led to faster communication convergence, and penalizing complexity created more human-like communication (supporting our two hypotheses). Further, we found that our novel neural architecture, VQ-VIB, achieved greater utility for the same complexity than other discrete methods. VQ-VIB combines the strengths of the methods it builds upon: it has a symbolic structure, defined by quantized vectors, but it is also explicitly uses an IB encoder to sample in a continuous space. Natural extensions of our work include further improvements to neural architectures, incorporation of new, cognitively-inspired terms to guide communication, or more complex domains, which could better illustrate advantages of VQ-VIB.

Our work represents a step towards reconciling principles from cognitive science with emergent communication literature, which we hope better supports human preferences. However, we recognize that, by quantifying properties of languages, our results may be misinterpreted to assign relative values to one language over another (e.g., higher complexity being better). We insist that our framework 1) has only been applied to narrow domains that do not capture any language's entirety and 2) does not prioritize any term and that any tradeoff between these terms is equally valid.

References

[1] Luc Steels and Tony Belpaeme. Coordinating perceptually grounded categories through language: A case study for colour. *Behavioral and Brain Sciences*, 28(4):469–489, 2005.

[2] Jakob Foerster, Ioannis Alexandros Assael, Nando de Freitas, and Shimon Whiteson. Learning to communicate with deep multi-agent reinforcement learning. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/file/c7635bfd99248a2cdef8249ef7bfbef4-Paper.pdf.

[3] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *Neural Information Processing Systems (NIPS)*, 2017.

[4] Igor Mordatch and Pieter Abbeel. Emergence of grounded compositional language in multi-agent populations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[5] Angeliki Lazaridou, Alexander Peysakhovich, and Marco Baroni. Multi-agent cooperation and the emergence of (natural) language. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL https://openreview.net/forum?id=Hk8N3Sclg.

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

[7] Tom Eccles, Yoram Bachrach, Guy Lever, Angeliki Lazaridou, and Thore Graepel. Biases for emergent communication in multi-agent reinforcement learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/fe5e7cb609bdbe6d62449d61849c38b0-Paper.pdf.

[8] Rahma Chaabouni, Eugene Kharitonov, Emmanuel Dupoux, and Marco Baroni. Communicating artificial neural networks develop efficient color-naming systems. *Proceedings of the National Academy of Sciences*, 118(12):e2016569118, 2021. doi: 10.1073/pnas.2016569118. URL https://www.pnas.org/doi/abs/10.1073/pnas.2016569118.

[9] Noga Zaslavsky, Charles Kemp, Terry Regier, and Naftali Tishby. Efficient compression in color naming and its evolution. *Proceedings of the National Academy of Sciences*, 115(31): 7937–7942, 2018. doi: 10.1073/pnas.1800521115. URL https://www.pnas.org/doi/abs/10.1073/pnas.1800521115.

[10] Noga Zaslavsky, Terry Regier, Naftali Tishby, and Charles Kemp. Semantic categories of artifacts and animals reflect efficient coding. In *Proceedings of the 41st Annual Meeting of the Cognitive Science Society*, 2019.

[11] Noga Zaslavsky, Mora Maldonado, and Jennifer Culbertson. Let's talk (efficiently) about us: Person systems achieve near-optimal compression. In *Proceedings of the 43rd Annual Meeting of the Cognitive Science Society*, 2021.

[12] Noga Zaslavsky, Karee Garvin, Charles Kemp, Naftali Tishby, and Terry Regier. The evolution of color naming reflects pressure for efficiency: Evidence from the recent past. *Journal of Language Evolution*, 2022.

[13] Francis Mollica, Geoff Bacon, Noga Zaslavsky, Yang Xu, Terry Regier, and Charles Kemp. The forms and meanings of grammatical markers support efficient communication. *Proceedings of the National Academy of Sciences*, 118(49), 2021.

[14] Naftali Tishby, Fernando C. Pereira, and William Bialek. The Information Bottleneck method. In *Proceedings of the 37th Annual Allerton Conference on Communication, Control and Computing*, 1999.

[15] Gal Chechik, Amir Globerson, Naftali Tishby, and Yair Weiss. Information Bottleneck for Gaussian variables. *Journal of Machine Learning Research*, 6, 2005.

[16] Ohad Shamir, Sivan Sabato, and Naftali Tishby. Learning and generalization with the Information Bottleneck. *Theoretical Computer Science*, 411(29-30):2696–2711, 2010.

[17] Naftali Tishby and Noga Zaslavsky. Deep learning and the information bottleneck principle. In *2015 IEEE Information Theory Workshop (ITW)*, pages 1–5, 2015. doi: 10.1109/ITW.2015.7133169.

[18] Alex Alemi, Ian Fischer, Josh Dillon, and Kevin Murphy. Deep variational information bottleneck. In *ICLR*, 2017. URL `https://arxiv.org/abs/1612.00410`.

[19] Ravid Shwartz-Ziv and Naftali Tishby. Opening the black box of deep neural networks via information. *arXiv preprint arXiv:1703.00810*, 2017.

[20] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[21] Mycal Tucker, Huao Li, Siddharth Agrawal, Dana Hughes, Katia P. Sycara, Michael Lewis, and Julie Shah. Emergent discrete communication in semantic spaces. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021. URL `https://openreview.net/forum?id=EuvW3-Lmoxm`.

[22] Angeliki Lazaridou, Karl Moritz Hermann, Karl Tuyls, and Stephen Clark. Emergence of linguistic communication from referential games with symbolic and pixel input. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL `https://openreview.net/forum?id=HJGv1Z-AW`.

[23] Kalesha Bullard, Franziska Meier, Douwe Kiela, Joelle Pineau, and Jakob Foerster. Exploring zero-shot emergent communication in embodied multi-agent populations. *arXiv preprint arXiv:2010.15896*, 2020.

[24] Kalesha Bullard, Douwe Kiela, Joelle Pineau, and Jakob Foerster. Quasi-equivalence discovery for zero-shot emergent communication. *arXiv preprint arXiv:2103.08067*, 2021.

[25] Matti Miestamo. On the feasibility of complexity metrics. In *FinEst linguistics, proceedings of the annual Finnish and Estonian conference of linguistics, Tallinn*, pages 11–26, 2004.

[26] Mathieu Rita, Rahma Chaabouni, and Emmanuel Dupoux. "LazImpa": Lazy and impatient neural agents learn to communicate efficiently. In *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 335–343, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.conll-1.26. URL `https://aclanthology.org/2020.conll-1.26`.

[27] Michael Cogswell, Jiasen Lu, Stefan Lee, Devi Parikh, and Dhruv Batra. Emergence of compositional language with deep generational transmission. *arXiv preprint arXiv:1904.09067*, 2019.

[28] Yi Ren, Shangmin Guo, Matthieu Labeau, Shay B. Cohen, and Simon Kirby. Compositional languages emerge in a neural iterated learning model. In *International Conference on Learning Representations*, 2020. URL `https://openreview.net/forum?id=HkePNpVKPB`.

[29] Łukasz Kuciński, Tomasz Korbak, Paweł Kołodziej, and Piotr Miłoś. Catalytic role of noise and necessity of inductive biases in the emergence of compositional communication. *Advances in Neural Information Processing Systems*, 34:23075–23088, 2021.

[30] Igor Mordatch and Pieter Abbeel. Emergence of grounded compositional language in multi-agent populations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), Apr. 2018. URL `https://ojs.aaai.org/index.php/AAAI/article/view/11492`.

[31] Rundong Wang, Xu He, Runsheng Yu, Wei Qiu, Bo An, and Zinovi Rabinovich. Learning efficient multi-agent communication: An information bottleneck approach. In *ICML*, pages 9908–9918, 2020. URL `http://proceedings.mlr.press/v119/wang20i.html`.

[32] Natasha Jaques, Angeliki Lazaridou, Edward Hughes, Caglar Gulcehre, Pedro Ortega, DJ Strouse, Joel Z Leibo, and Nando De Freitas. Social influence as intrinsic motivation for multi-agent deep reinforcement learning. In *International Conference on Machine Learning*, pages 3040–3049. PMLR, 2019.

[33] Toru Lin, Jacob Huh, Christopher Stauffer, Ser Nam Lim, and Phillip Isola. Learning to ground multi-agent communication with autoencoders. *Advances in Neural Information Processing Systems*, 34, 2021.

[34] Mike Dowman. Explaining color term typology with an evolutionary model. *Cognitive Science*, 31(1):99–132, 2007.

[35] Mikael Kågebäck, Emil Carlsson, Devdatt Dubhashi, and Asad Sayeed. A reinforcement-learning approach to efficient communication. *PLOS ONE*, 15(7):1–26, 07 2020.

[36] David Lewis. *Convention: A philosophical study*. John Wiley & Sons, 2008.

[37] Brent Berlin and Paul Kay. *Basic Color Terms: Their Universality and Evolution*. University of California Press, Berkeley and Los Angeles, 1969.

[38] Paul Kay, Brent Berlin, Luisa Maffi, William R. Merrifield, and Richard Cook. *The World Color Survey*. Stanford: Center for the Study of Language and Information, 2009.

[39] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[40] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. 2016.

[41] Noga Zaslavsky and Naftali Tishby. Deterministic annealing and the evolution of optimal information bottleneck representations. 2019.

[42] Will Monroe, Robert X.D. Hawkins, Noah D. Goodman, and Christopher Potts. Colors in context: A pragmatic neural model for grounded language understanding. *Transactions of the Association for Computational Linguistics*, 5:325–338, 2017.

[43] Jennifer Hu, Roger Levy, and Noga Zaslavsky. Scalable pragmatic communication via self-supervision. In *ICML Workshop on Self-Supervised Learning for Reasoning and Perception*, 2021. URL `https://arxiv.org/abs/2108.05799`.

[44] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax, 2017.

[45] Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. In *International Conference on Learning Representations*, 2017. URL `https://openreview.net/forum?id=S1jE5L5gl`.

[46] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256, 1992.

Checklist

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default **[TODO]** to [Yes] , [No] , or [N/A] . You are strongly encouraged to include a **justification to your answer**, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes] See Section TODO.
- Did you include the license to the code and datasets? [No] The code and the data are proprietary.
- Did you include the license to the code and datasets? [N/A]

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See abstract, introduction, and findings
   (b) Did you describe the limitations of your work? [Yes] See Section 6, in which we note that we tested in limited domains and Section 3.2.4, in which we note that an approximation we made may be improved upon in future work.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Section 6 in which we discuss how readers may misconstrue quantitative measures of languages as assigning value to one over the other.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See footnote for link to anonymized code
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Appendix A
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See error bars in plots and Table 1, in Section 5. All scatter plots directly included the raw results from all 5 random trials.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] Training times and computational resources are listed in Appendix A.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] We use the World Color Survey (WCS) data, as well as the publicly-available code by Zaslavsky et al. [9]. Both sources are cited.
   (b) Did you mention the license of the assets? [Yes] We note that both are publicly-available and link to them in Footnote 1.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] Anonymized code is linked to in a footnote.

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A] We only used the existing WCS dataset.

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A] We only used the existing WCS dataset.

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Appendix: Implementation Details

Here, we include implementation details that we omitted from the main paper that can be used to reproduce our experiments.

A.1 Multi-Agent Reinforcement Learning Experiments

We trained agents in the 9 points and Uniform environments with Multi-Agent Deep Deterministic Policy Gradient (MADDPG), a standard method for multi-agent reinforcement learning [3].

Agent architectures were identical in both environments. A speaker agent consisted of 2, 64-neuron fully-connected layers with ReLU activations, followed by a speaker "head." The head implemented the specific communication method (e.g., continuous, VQ-VIB, etc.) by taking the 64-dimensional representation from the speaker and going through one of 5 possible architectures to produce a vector in the communication space, R^c. (As discussed later, c varied by environment and architecture slightly.)

- **Continuous:** The continuous communication head consisted of two fully connected layers branching off the speaker encoding to generate a μ and `logvar` (the log of the variance) from which the communication vector was sampled using the reparametrization trick. Complexity was measured as the KL divergence between the parameters μ and `logvar` and a unit Gaussian.
- **Onehot:** The onehot communication head consisted of one fully connected layer from the speaker encoding to R^c. Onehot vectors were generated using gumble softmax sampling, with temperature 1. Complexity was measured as the KL divergence between the distribution from which the onehot vector was sampled and a uniform categorical distribution. Thus, complexity corresponds to the difference between the conditional and unconditional entropies of messages, unlike non-IB frameworks that often penalize unconditional entropy.
- **VQ-VIB:** The VQ-VIB communication head first passed the speaker encoding through two fully connected layers, branching off the encoding, to generate a μ and `logvar`, as in the continuous communication case. A communication vector was sampled from the distribution, whereupon it was discretized to the nearest quantized vector, ζ, in the communication space (as done in a standard VQ-VAE, without the sampling). Complexity was measured as the KL divergence between the parameters μ and `logvar` and a unit Gaussian.
- **VQ-VIB After:** The VQ-VIB After communication head first passed the speaker encoding through two fully connected layers to generate an encoding, z, and `logvar`. The encoding, z, was discretized to the nearest quantized vector, ζ. Lastly, a communication vector, c, was sampled from a normal distribution centered at ζ with variance specified via `logvar`. Complexity was measured as the KL divergence between the parameters ζ and `logvar` and a unit Gaussian.

- **Proto:** The Proto head mapped the speaker encoding through a single fully-connected layer to a categorical distribution over k neurons, for k representing the number of prototypes. A k-dimensional onehot vector was sampled using gumble softmax; a prototype, p was selected by multiplying the prototype matrix by the onehot vector. A different fully-connected layer mapped from the speaker encoding to `logvar`. Lastly, a communication vector, c, was sampled from a Gaussian centered at p with variance via `logvar`. Complexity was measured as the KL divergence between the parameters p and `logvar` and a unit Gaussian. This closely resembles the VQ-VIB After architecture, but with a different discretization mechanism.

All agents were trained with an Adam optimizer with learning rate 0.01 and batch size 1024.

A.1.1 9 points Environment

For all agent architectures except for onehot, the communication dimension was 2; onehot used communication dimension 9 so that the number of discrete representations allowed by the architecture was not the limiting factor for complexity. Similarly, the VQ-VIB and Proto agents used 9 discrete embeddings.

In our main results (presented in Figure 4), agents were trained for 10,000 episodes, with each episode lasting 100 timesteps. The listener agent was always spawned at the origin, equidistant from each landmark. The informativeness loss weight, λ_I, was constant at 1.0. For the continuous and VQ-VIB communication methods, the complexity loss weight, λ_C increased by $8 * 10^{-6}$ per episode. The onehot and Proto methods suffered from communication collapse at that rate, so results λ_C increased half as quickly: at $4 * 10^{-6}$ per episode.

Training a team in the 9 points environment for 10,000 episodes took approximately 15 minutes on a desktop computer with 16 Intel i9 cores for all architectures, except for onehot, which took roughly 20 minutes. This difference in time is attributable to the larger communication dimension needed for onehot.

A.1.2 Uniform Environment

In the Uniform environment, we used identical parameters as in the 9 points environment, except for the changes noted below.

For all agent architectures except for onehot, the communication dimension was 3; for onehot, we experimented with communication dimensions of 9 and 100. Similarly, we tested VQ-VIB and Proto agents when using 9 or 100 discrete embeddings. The listener agent was spawned at a starting location sampled uniformly at random from the map (bounded between -200 and 200 for both x and y). Otherwise, we used the same number of training episodes, episode duration, and complexity annealing rates as in 9 points.

Training a team in the Uniform environment for 10,000 episodes took approximately 15 minutes, except for onehot, which took roughly 30 minutes for 100-dimensional communication.

A.2 Color Reference Game Experiments

Here, we include the implementation details used in our color reference game experiments.

Agents were trained in the overall architecture depicted in Figure 7. The speaker agent was parametrized with 3 fully connected layers with ReLU activations and hidden dimension 64, followed by a communication head. The communication heads used the same architectures as specified in the particle world experiments: gumbel softmax was used for generating onehot vectors, and a straight-through estimator was used for discretization in VQ-VIB architectures [44, 45]. In all experiments, the communication dimension was set to 32 for all agents except onehot, for which the communication dimension was set to 330 (to match the expressive power of other discrete agents, that each had 330 discrete tokens in the 32 dimensional space).

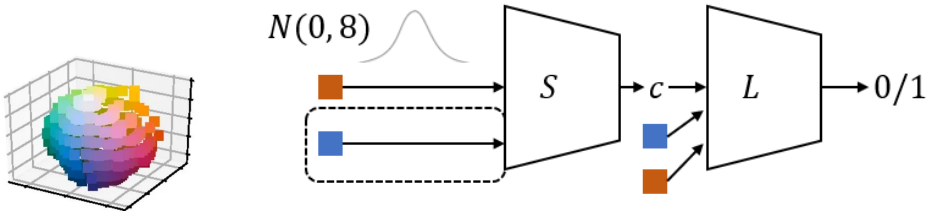
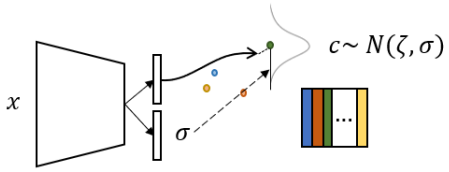


Figure 7: Target and distractor colors were drawn uniformly without replacement from the set of 330 colors in CIELAB space (left). A speaker, S, observed the target and (in the Common Ground setting) distractor, corrupted by Gaussian noise. The listener, L, had to identify which of the shuffled colors it saw was the target color, given the speaker's communication, c.



$$\zeta = \text{argmin}_{\zeta}[(z(x) - \zeta)^2]$$
$$q_{SA}(c|x) = \mathcal{N}(\zeta, \sigma(x)) \tag{4}$$

Figure 8: In the VQ-VIB After architecture (a), the speaker deterministically maps x to the nearest quantized encoding, ζ, and then c is sampled by adding Gaussian noise to ζ.

Without annealing the complexity loss, 50 training epochs sufficed for agents to converge to a stable solution. Thus, in our experiments without complexity loss, we only trained for 50 epochs. For later experiments, wherein we annealed complexity, we first trained agents for 50 epochs before beginning to anneal the complexity term for the next 40 epochs (for a total of 90 epochs). We note that, for VQ-VIB, we set the complexity loss to 0 during this "burnin" period, but for onehot communication, it appeared necessary to set the loss to 0.05 to prevent numerical instability during the first epoch after the burnin period. During complexity loss annealing, we increased λ_C by 0.01 during each epoch for VQ-VIB, and by 0.1 for onehot. These different rates were necessary to induce similar ranges in complexity of learned communication.

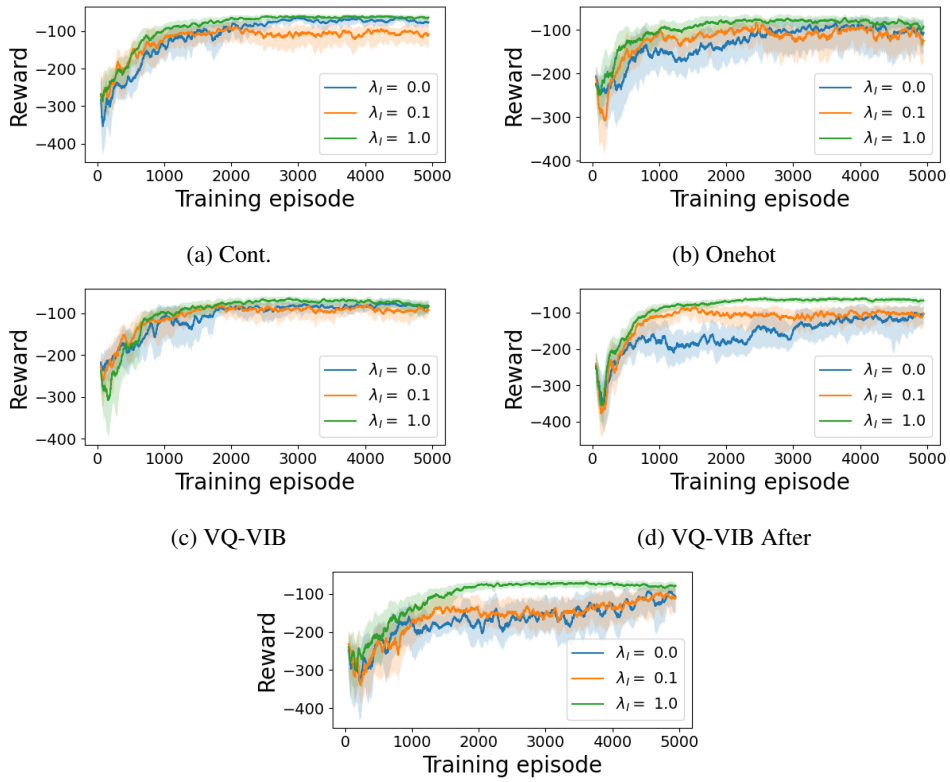
In the Isolated color reference game, the speaker observed a only the 3-dimensional, corrupted target color. In the Common Ground setting, the speaker observed a 6-dimensional vector, generated by concatenating noisy versions of the target and distractor colors. In both settings, the listener observed the communication output by the speaker, as well as the target and distractor colors (ordered randomly, without added noise). Results were analyzed using code partially-based on publicly-available code released by Zaslavsky et al. [9].

We trained agents on a desktop computer with 16 Intel i9 CPUs and 1 NVIDIA GeForce RTX 2080. Training for 90 epochs in both the Isolated and Common Ground settings took approximately 3 minutes for all agent architectures. Agents were trained with an Adam optimizer with default parameters, with batch size 1024.

B Appendix: VQ-VIB After

We proposed an additional architecture, **VQ-VIB After**, that is closely related to our main VQ-VIB method. In VQ-VIB After, an encoder to generates a discrete representation ζ, whereupon a communication vector c is sampled from a Gaussian in the continuous space centered around ζ (Equation 4). This "sample-after" method is depicted in Figure 8, and corresponds to an intuitive notion of choosing a word (a discrete embedding) deterministically but then being uncertain about the form (by sampling in the continuous space around it).

This architecture fits into the standard Information Bottleneck framework, as complexity can be penalized via standard variational methods. However, it is distinct from discrete communication



(a) Cont.

(b) Onehot

(c) VQ-VIB

(d) VQ-VIB After

(e) Proto

Figure 9: In the 9 points environment, using a higher weight for informativeness (λ_I) resulted in faster convergence to higher rewards for each communication architecture. Medians and quartiles plotted over 5 trials for each configuration.

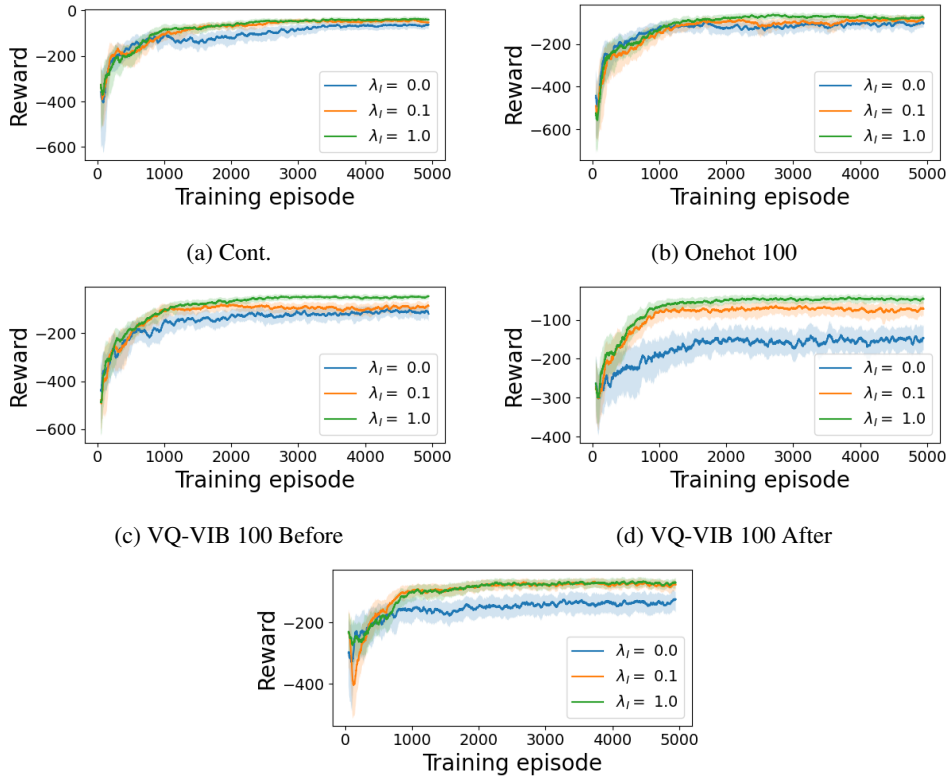
methods because communication vectors are ultimately sampled from a continuous space. Thus, this work does not fit naturally into the main focus of this work, but we believe it could have applications in other settings, such as speech processing (wherein discrete words are pronounced in a continuous auditory space).

C Appendix: Full Particle World Results

For brevity, we only included partial results from the particle-world experiments in the main paper. Here, we include the full results.

First, we experimented with varying λ_I for different environments and neural architectures, without annealing complexity. Training curves for both environments, using varying λ_I for each neural architecture, are included in Figures 9 and 10. For each configuration (a particular speaker architecture and λ_I), we trained 5 teams from scratch for 5,000 episodes and plotted the median and quartile rewards, smoothed over a 50 episode window.

A consistent trend emerged: higher λ_I led to faster convergence to a higher reward. Indeed, this corroborates findings by Lin et al. [33], who used an autoencoder architecture for communication in multiagent settings, but here we showed good convergence properties when mixing utility and informativeness terms (whereas they reported worse results when adding utility losses). Overall, these findings motivated training agents to convergence with high λ_I before beginning to penalize complexity with λ_C.



(a) Cont.

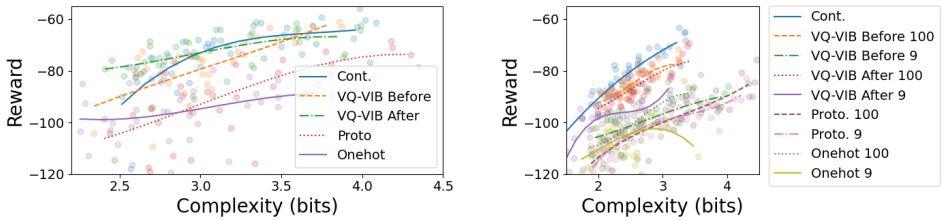
(b) Onehot 100

(c) VQ-VIB 100 Before

(d) VQ-VIB 100 After

(e) Proto 100

Figure 10: In the Uniform environment, we observed similar trends as in 9 points: greater λ_I improved convergence rates and rewards. Medians and quartiles plotted over 5 runs.

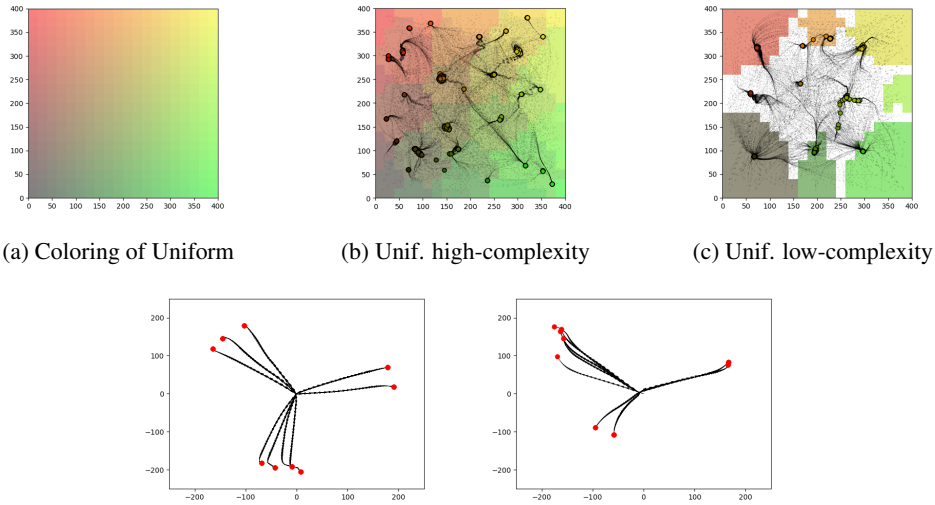


(a) 9 points: Reward vs. Complexity

(b) Uniform: Reward vs. Complexity

Figure 11: In the 9-points and Uniform environments, agent reward (utility) decreased with complexity. Each point represents the mean complexity and reward for agent checkpoints at 1,000 episode increments during training, over 5 random seeds. Curves are the best cubic fits to the data. VQ-VIB architectures typically outperformed other discrete methods by achieving higher reward for the same complexity, and over-parametrizing with additional embeddings (100 vs. 9) also helped.

Next, we include the results of annealing complexity in both the 9 points and Uniform environments, as depicted in Figure 11. In both environments, as we increased λ_C, complexity and reward decreased. Comparing both VQ-VIB architectures to other discrete communication methods showed that VQ-VIB agents tended to obtain a higher mean reward, for a given complexity, than other methods. In the Uniform environment, we also compared different variants of VQ-VIB, Proto, and onehot agents based on the maximum number of discrete messages allowed by the neural architectures. We tested agents parametrized with either 100 or 9 unique messages; we hypothesized that initializing agents with many discrete messages and then gradually decreasing complexity would result in better



(a) Coloring of Uniform (b) Unif. high-complexity (c) Unif. low-complexity

(d) 9 points high-complexity (e) 9 points low-complexity

Figure 12: Visualizations of different discretizations in the particle world experiments. In the Uniform environment (top row), the speaker used many or few tokens to refer to fine or coarse discretizations of the continuous space when using high or low complexity communication (respectively). Each location was associated with a unique color, depicted in (a). Such coloring was not applicable in the 9 points environment (bottom row). In both environments, we traced listener trajectories (black) and final locations (dots). As complexity decreased, the number of distinct end locations decreased, indicating coarser discretizations of the space.

performance than training with few messages from the start. Indeed, we observed that was the case. As seen in Figure 11 b, VQ-VIB agents parametrized with 100 embeddings consistently achieved higher reward for the same complexity than VQ-VIB agents parametrized with only 9 embeddings. This trend held even when the complexity was lower than $\log_2(9)$ bits.

Lastly, we visually analyzed VQ-VIB communication in the 9 points and Uniform environments. As agents used less complex communication (i.e., fewer distinct messages from the speaker to the listener), one would expect each message to refer to coarser regions of space and for the listener to therefore end in a smaller set of possible destinations. We recorded, over 1,000 episodes, the target location, the speaker's communication, and the listener's trajectory. In the Uniform environment, using a discretized representation of the continuous space (for visualization purposes), we performed similar "modemap" analysis as in the color reference game. That is, we recorded the most likely token associated with each target location and colored all locations associated with the same target with the same color. (The exact color was determined via a color map designed for visualization purposes, or left as white if no single token was used for that location more than 50% of the time.) In the 9 points environment, we did not perform such coloring because the set of possible destinations was already discrete. In both environments, however, we visualized listener agent's trajectories. Visualizations of VQ-VIB teams are included in Figure 12.

Figure 12 is consistent with expected behavior. As communication complexity decreased, the speaker agent used fewer distinct tokens and the listener moved to fewer distinct destinations. Furthermore, as seen in the Uniform environment, the speaker communicated about increasingly coarse locations (indicated by larger spatial regions referred to by the same token, visualized with color). This visual analysis confirms similar trends as in the color domain, highlighting the general applicability of our framework for shaping communication complexity.

D Appendix: Color Reference Game Full Results

We omitted some of the color reference game results from the main paper due to space constraints. Here, we include our complete results.

D.1 Isolated Color Reference Game

First, we include results from the Isolated setting, in which we trained agent with different values of λ_I without penalizing complexity. These results are included in Figure 13. Setting $\lambda_I = 0$ induced similar conditions to those used by Chaabouni et al. [8]; any positive λ_I favored more informative (and therefore more complex) communication. Given the monotonically increasing nature of the IB bound, an optimal agent trained with any positive λ_I would maximize utility and informativeness by maximizing complexity, up to a maximum of $\log_2(330)$ bits. Instead, we found that agents appeared to converge to lower complexity values, although increasing λ_I tended to somewhat increase complexity and informativeness. For example, onehot communication increased in complexity from roughly 2.3 bits for $\lambda_I = 0.0$ to 3.0 bits for $\lambda_I = 1.0$. This alone is a novel result compared to Chaabouni et al. [8], who trained agents with no informativeness loss and observed lower-complexity communication. Curiously, VQ-VIB agents automatically converged to informativeness around 3.5 bits regardless of the λ_I value, although increasing λ_I did improve training stability. This difference between onehot and VQ-VIB indicates that the differing architectures themselves have different inductive biases.

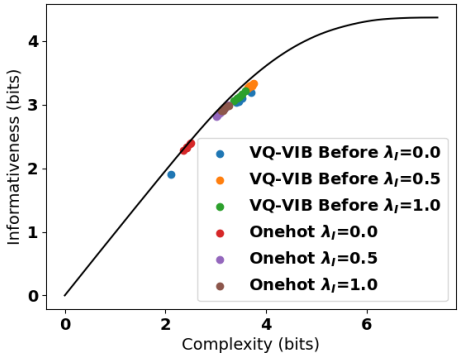
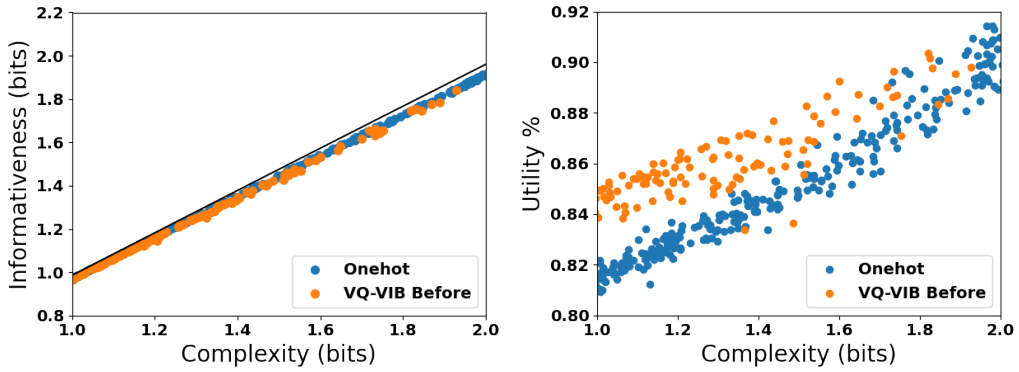


Figure 13: Training agents in the Isolated color reference game for various λ_I but no complexity loss. Without positive λ_I, onehot learns low-informativeness communication, but increasing λ_I generated more complex agents, especially for onehot.

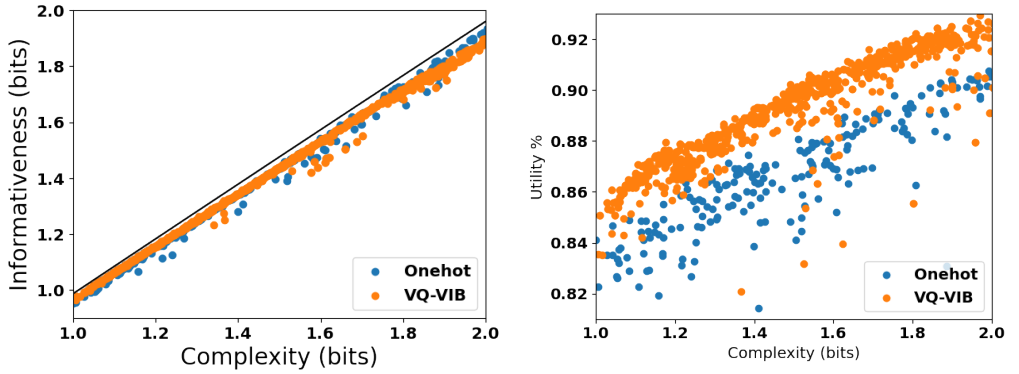
We conducted further experiments to compare agents with the neural agents presented by Chaabouni et al. [8], wherein onehot agents were parametrized with a communication dimension of 1024. We therefore tested VQ-VIB and onehot agents with 1024 tokens, instead of the 330 used for the results in our main paper.team This hyperparameter change largely did not alter results. As shown in Figure 14, both VQ-VIB and onehot achieved nearly optimal informativeness for a given complexity, and VQ-VIB achieved higher utility than onehot (results zoomed in to complexity values between 1 and 2 bits for clearest plotting of these trends).

Lastly, we performed additional experiments to investigate the effect of training with REINFORCE, instead of the supervised learning framework used in our main experiments (e.g., backpropagating through gumbel softmax) [46]. Chaabouni et al. [8] had found that training with REINFORCE typically resulted in lower-complexity communication than when training with messages generated via a gumbel softmax layer, and that training with REINFORCE was less stable. Results from our REINFORCE experiments, using 330 tokens for onehot and VQ-VIB, are included in Figure 15. We trained agents with $\lambda_U = 1.0$, $\lambda_I = 50.0$, and λ_C was annealed from 0 to 0.1, and our code was adapted from Rita et al. [26].



(a) (b)

Figure 14: Communicative informativeness (a) and utility (b) as a function of complexity for color agents trained with 1024 tokens in the Isolated game. These results largely reproduced results from our main paper with 330 tokens, wherein VQ-VIB and onehot were similarly efficient, but VQ-VIB achieved higher utility than onehot.



(a) (b)

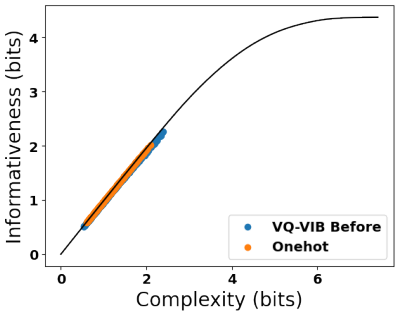
Figure 15: Informativeness (a) and utility (b) versus complexity for agents trained with 330 tokens, using REINFORCE. These results closely tracked prior experiments: both onehot and VQ-VIB were efficient, and VQ-VIB achieved greater utility, for the same complexity, than onehot.

Our results from these experiments align with all prior findings: VQ-VIB and onehot achieved high communicative efficiency, and VQ-VIB produced greater utility than onehot did. Beyond those corroborating results, our experiments aligned with Chaabouni et al. [8]'s findings and further illustrated the importance of informativeness losses and the VQ-VIB architecture. With zero informativeness loss (i.e., $\lambda_I = 0$) onehot agents often failed to converge to any useful communication (as indicated in prior art), but by increasing λ_I, agents consistently converged to useful communication. Indeed, with $\lambda_I = 50$ and $\lambda_C = 0$, onehot agents learned to communication with complexity of up to 2.4 bits, greater than any complexity value found by Chaabouni et al. [8] using REINFORCE over 180 seeds (see Fig. 5 in that paper). Furthermore, we observed that VQ-VIB agents typically converged to higher informativeness (and higher complexity) communication than onehot when complexity was not penalized, indicating that further investigation of inductive biases is a promising direction for future work. Overall, this experiment largely replicates our findings from the main paper.

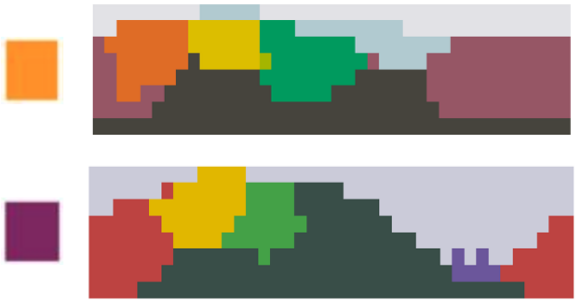
Thus, findings from our REINFORCE experiments indicate that the main conclusions of our paper hold true across several training mechanisms, and furthermore that our framework (including in particular the informativeness loss) can address issues already noted in prior work.

D.2 Common Ground Color Reference Game

Here, we include the full results from our Common Ground color reference game experiments. Given the lack of human-labeled data for such a setting, we were unable to compare agent communication with human languages. However, by analyzing communication when varying the target for a fixed distractor, or by varying the distractor for a fixed target, we established that agents conditioned communication both on the target and distractor colors.



(a) Informativeness and Complexity for a gray distractor



(b) Communication by the same agent conditioned on purple or orange distractors differed, showing context-dependent communication.

Figure 16: In Common Ground experiments, both VQ-VIB and onehot communication demonstrated high informativeness for a fixed distractor (left), and varied communication according to distractor (right).

We trained agents in the Common Ground setting by randomly sampling target and distractor colors, as in the Isolated setting, but here the speaker observed (noisy) versions of the distractor as well as the target. Based on insights that training to increase informativeness produced more complex communication, we trained agents for 30 episodes with $\lambda_I = 1.0$. For the next 70 episodes, we annealed the complexity weight, λ_C, using the same rates as in the Isolated setting; this produced communication at different levels of complexity. In the main paper, we showed how VQ-VIB achieved higher utility, for the same complexity, than onehot (Figure 6).

We further analyzed communication in two ways: using a fixed distractor with varying targets and using a fixed target with varying distractors. First, for a fixed, gray distractor (#46 in the WCS data), we measured the standard metrics of informativeness and complexity. That is, we set the distractor color to gray, and we recorded, for each of the other 329 colors as the target, the speaker's communication. We then performed the same complexity and informativeness analysis as in the Isolated setting, and found that both VQ-VIB and onehot communication remained near-optimally efficient, as shown in Figure 16 a.

Second, we measured whether agents conditioned their communication on the distractor color. Unlike the prior evaluation, in which we varied the target color for a fixed distractor, here, we evaluated communication for different distractors. Plots of a VQ-VIB agent's naming scheme, using either an orange (color #212) or purple (color #325) distractor are shown in Figure 16 b.

As expected, the agent displayed different resolutions in regions of color depending upon the distractor. Specifically, for an orange distractor, the speaker had finer resolution around shades of orange and yellow; conversely, the speaker had finer resolution around shades of purples for a purple distractor. This aligns with expectations that, when trying to differentiate between colors, one must use more complex or accurate words to distinguish similar colors.

Jointly, our findings indicate that many of the themes identified in the Isolated setting – nearly-optimal communicative efficiency, the importance of annealing λ_C to generate varied complexity, and the advantages of VQ-VIB over onehot – held in the Common Ground setting as well.