Implicit Repair with Reinforcement Learning in Emergent Communication

Fábio Vital
INESC-ID & Instituto Superior Técnico
Lisboa, Portugal

Alberto Sardinha
INESC-ID & PUC-Rio
Rio de Janeiro, Brazil

Francisco S. Melo
INESC-ID & Instituto Superior Técnico
Lisboa, Portugal

ABSTRACT

Conversational repair is a mechanism used to detect and resolve miscommunication and misinformation problems when two or more agents interact. One particular and underexplored form of repair in emergent communication is the implicit repair mechanism, where the interlocutor purposely conveys the desired information in such a way as to prevent misinformation from any other interlocutor. This work explores how redundancy can modify the emergent communication protocol to continue conveying the necessary information to complete the underlying task, even with additional external environmental pressures such as noise. We focus on extending the signaling game, called the Lewis Game, by adding noise in the communication channel and inputs received by the agents. Our analysis shows that agents add redundancy to the transmitted messages as an outcome to prevent the negative impact of noise on the task success. Additionally, we observe that the emerging communication protocol's generalization capabilities remain equivalent to architectures employed in simpler games that are entirely deterministic. Additionally, our method is the only one suitable for producing robust communication protocols that can handle cases with and without noise while maintaining increased generalization performance levels. Our code and appendix are available at https://fgmv.me/projects/noisy-emcom. First author correspondence: fabiovital@tecnico.ulisboa.pt.

KEYWORDS

emergent communication; representation learning; reinforcement learning; multi-agent reinforcement learning

ACM Reference Format:
Fábio Vital, Alberto Sardinha, and Francisco S. Melo. 2025. Implicit Repair with Reinforcement Learning in Emergent Communication. In *Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Detroit, Michigan, USA, May 19 – 23, 2025*, IFAAMAS, 41 pages.

1 INTRODUCTION

Emergent Communication (EC) is a field that recently gained attention in Machine Learning (ML) research. The progress of language evolution research [4, 5, 9, 43, 51, 53] and the conceptualization of artificial languages for robot and human-robot communication [3, 17, 23, 34] are some of the fundamental motivations behind the recent rise in interest. Mainly, EC focuses on developing experiments where a group of agents must learn how to communicate without prior knowledge to achieve a common goal, where coordination and cooperation are essential for the group's success [29, 36, 55]. This approach differs from the current state of the art in natural language processing (NLP), where large language models (LLMs) dominate the field. LLMs are supervised statistical models that optimize the prediction of the next token given a context (textual input) [10, 12, 22, 38]. It is still an open question whether working only in the language space (as LLMs do) is enough to create agents with an intrinsic and deeper meaning about the world that are capable of adapting to novel circumstances effectively [2]. As such, we argue that exploring different approaches, like EC, is crucial to continue advancing the field of NLP.

The main focus of this work comprises the study of a specific topic in language evolution called *conversational repair* [46]. In linguistics, conversational repair is already a known topic that plays an important role in establishing complex and efficient communication protocols [1]. In short, conversational repair aggregates any communication mechanism employed by any interlocutor to initiate a process to detect and clarify some information being transferred by any other interlocutor. To give more context on how our work relates to previous literature, we divide repair mechanisms into two broad categories: implicit and explicit, a coarser partitioning of the one introduced by Lemon [28]. Explicit repair mechanisms happen when an interlocutor distinctly starts a follow-up interaction (communication) in order to clarify some conveyed past information, e.g., "Is it the blue one?" (confirmation); "Is it the first or the second one?" (clarification). On the other hand, implicit mechanisms happen in a subtle way where the interlocutor, conveying the original information, intentionally expresses it in such a way as to minimize misinformation and preemptively avoid posterior conversational repair phases altogether. The implicit conversational repair mechanism also has connections to the concept of redundancy in linguistics and communication analysis [8]. Redundancy appears in every human language, at the semantic and syntax level, and appears in the form of repetition or extra content to send. Additionally, the sending interlocutor may apply different levels of redundancy that she/he finds necessary to convey the information given the target audience and medium used for communication.

Most previous works in EC employ variations of a signaling game called the Lewis Game (LG) [29], with the primary purpose of analyzing how a communication protocol emerges as the result of achieving cooperation to solve the game [9, 16, 23]. In the LG, the Speaker describes an object to the Listener, who then has to discriminate it against a set of distractor objects. We call the union of the assigned object and distractors the *candidates*. Regarding this study, we extend a variation introduced by Chaabouni et al. [6], where real

Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Y. Vorobeychik, S. Das, A. Nowé (eds.), May 19 – 23, 2025, Detroit, Michigan, USA. © 2025 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

images are used as the objects to discriminate instead of categorical inputs, and the number of candidates given to the Listener increases in several orders of magnitude, conveying a more complex game than in previous works [4, 5, 26, 34, 44, 51]. In the original work, the authors propose a supervised training routine where the Listener receives the correct answer after each game. However, this implementation diverges from human communication, where there is usually no direct supervision on how effective a particular dialogue can be [18]. We propose modeling both agents as RL agents. As such, the only (semi) supervised information given to the agents is the outcome of the game. Similar to previous works [17, 26, 30, 44], we model both agents using Reinforce [54].

Furthermore, as a means to study implicit repair mechanisms, we define a new game variation with faulty communication channels that can introduce noise into the messages. This new game setup has the necessary conditions to study if agents can detect and overcome miscommunication/misinformation to solve the game cooperatively. Our analysis shows that the emerging communication protocols have redundancy built in to prevent the adverse effects of noise, where even partial messages have enough information for the Listener to select the correct candidate. Additionally, we show that the training in the noisy game produces communication protocols that are highly robust to noise, being effective in different noise levels, even without noise, at test time.

Previous literature has already addressed the problem of explicit conversational repair. Lemon [28] propose new research directions on how to embed conversational repair into EC tasks, where the repair mechanism acts as a catalyst to fix misalignments, for example, in the language learned by each agent for a specific cooperative task. Moreover, another recent work develops an extended version of the Lewis Game to enable a feedback mechanism from the Listener to the Speaker, mimicking the initialization of an external repair mechanism phase [35]. However, some limitations compromise the co-relation to human languages. First, the feedback sent by the Listener contains minimal information (single binary token), and such feedback is sent after every token in the message, breaking the turn-taking nature of the dialogue. Compared to our work, we designed a more challenging game where we prevent cyclic feedback (from the Listener to the Speaker), meaning the Speaker does not receive direct feedback about how noise affects the messages being transferred. In our case, the Speaker only knows the result of the game. Consequently, the Speaker needs to understand through trial and error how to convey information to facilitate the Listener's job, inducing an implicit repair mechanism, as explained previously.

To summarize, our contributions are 3-fold. First, although previous works introduce game designs featuring noisy channels, we contribute with a rigorous mathematical derivation on how to aggregate noise into the LG. We found such inference lacking in the literature. Furthermore, we define a new noisy game variant where the input objects to discriminate are injected with noise. We use this new variant as an out-of-distribution game to evaluate how the trained protocols react to new forms of noise. Secondly, we demonstrate the effectiveness of employing RL agents on complex LG variants featuring the discrimination of natural images and noisy communication channels. Additionally, we showcase that the RL variant achieves better results than the original architecture with a supervised Listener. We emphasize that our objective is not to benchmark different RL algorithms but to show that implementing the Listener as an RL agent can bring advantages against the Supervised counterpart, where even a straightforward implementation of Reinforce is enough to observe substantial gains already. Third, we analyze and show how more complex game designs, such as introducing noise to the LG, guide the agents to resort to redundancy measures to complete the game efficiently, mimicking implicit conversational repair mechanisms. We show how the protocols emerging to communicate through noisy channels have better generalization capabilities and robustness to different noise levels at test time. Additionally, we illustrate that these improvements are a side-effect of resorting to redundancy in the messages sent.

2 METHODOLOGY

We start this section by defining a noisy variation of the LG ,called the Noisy Lewis Game (NLG). The main change in the NLG incorporates a faulty communication channel where noise interferes with the transmitted messages by masking a subset of the tokens. This game variation is more complex than the LG, where the (RL) game environment becomes stochastic. We further note that the original LG is a simplification of the NLG where we fix the noise level at zero. Afterward, we detail how the Speaker converts the received input into a message, a sequence of discrete tokens, and how the Listener processes the message and candidates to make decisions. We impose a RIAL setting [13], where agents are independent and perceive the other as part of the environment. Hence, we describe the learning strategy for both agents independently, explaining the loss composition and the importance of each loss term to guide training where functional communications protocols can emerge.

2.1 Noisy Lewis Game (NLG)

The Noisy Lewis Game (NLG) is a discrimination game in which one of the agents, the *Speaker*, must describe an object by sending a message to the other agent, the *Listener*. When the game starts, the Speaker receives a target image $\boldsymbol{x}$ retrieved from a fixed dataset $\boldsymbol{x} \in \mathbb{X}$ and describes it by generating a message, $m : \mathbb{X} \times \mathbb{R}^K \rightarrow \mathbb{W}^N$, where $\mathbb{W}$ is a finite vocabulary, and $\boldsymbol{\theta} \in \mathbb{R}^K$ parametrizes m. The message comprises N discrete tokens, $m(\boldsymbol{x}; \boldsymbol{\theta}) = (m_t(\boldsymbol{x}; \boldsymbol{\theta}))_{t=1}^N$, where $m_t(\boldsymbol{x}; \boldsymbol{\theta}) \in \mathbb{W}$. Due to the noisy nature of the communication channel, the Listener can receive a message with unexpected modifications. We model this perturbation with the function $n : \mathbb{W}^N \rightarrow \mathbb{W}'^N$, where the function processes each token independently and converts it into a default unknown token with a given probability. As such, $\mathbb{W}'$ is the union of the original vocabulary plus the unknown token, $\mathbb{W}' = \mathbb{W} \cup \{\mathsf{unk}\}$. We describe introduce n as :

$$n(m(\boldsymbol{x}; \boldsymbol{\theta})) = (n_t(m_t(\boldsymbol{x}; \boldsymbol{\theta})))_{t=1}^N$$
$$\text{s.t. } n_t(m_t(\boldsymbol{x}; \boldsymbol{\theta})) = \begin{cases} m_t(\boldsymbol{x}; \boldsymbol{\theta}), & \text{if } \mathrm{p} > \lambda \\ \mathsf{unk}, & \text{otherwise,} \end{cases} \tag{1}$$

where p is sampled from a uniform distribution, $\mathrm{p} \sim \boldsymbol{\mathcal{U}}(0, 1)$, and $\lambda \in [0, 1)$ is a fixed threshold, indicating the noise level present in the communication channel. By definition, the Speaker is agnostic to this process and will never know if the message was modified. For simplicity, we define $\boldsymbol{m}$ to describe a (noisy) message given some input, $\boldsymbol{m} = m(\boldsymbol{x}, \boldsymbol{\theta})$ or $\boldsymbol{m} = n(m(\boldsymbol{x}, \boldsymbol{\theta}))$. We also refer each message token as m_t instead of $m_t(\boldsymbol{x}; \boldsymbol{\theta})$, omitting the domain.

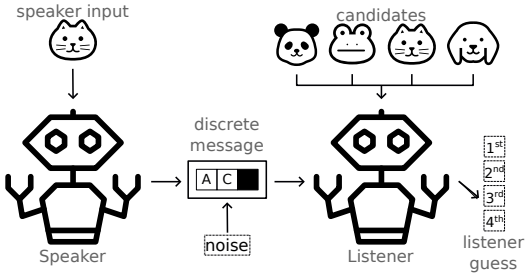

Figure 1: Visual Representation of the Noisy Lewis Game (NLG). In this illustration, the message, m, contains three tokens ($N = 3$), where the last one is masked.

Subsequently, the Listener receives the message along with a set of candidate images, $\mathbb{C} \in [\mathbb{X}]^C$, where $[\mathbb{X}]^C$ defines the set of all subsets with C elements from $\mathbb{X}$. With both inputs, the Listener tries to identify the image the Speaker received, $\boldsymbol{x}$. We define choice : $\mathbb{W}'^N \times [\mathbb{X}]^C \times \mathbb{R}^{k'} \to \mathbb{J}$ to specify the Listener's discrimination process, where $\mathbb{J} \subset \mathbb{N}$ is a particular enumeration of $\mathbb{C}$, such that $\mathbb{C} = \bigcup_{j \in \mathbb{J}} \mathbb{C}_j$, $|\mathbb{J}| = C$, and $\boldsymbol{\phi} \in \mathbb{R}^{k'}$ parametrizes choice. Therefore, the index outputted by the Listener, $j = \text{choice}(\boldsymbol{m}, \mathbb{C}; \boldsymbol{\phi})$, is the j-th element in $\mathbb{C}$, denoting the final guess $\hat{\boldsymbol{x}} = \mathbb{C}_j$.

Both agents receive a positive reward if the Listener correctly identifies the target image $\boldsymbol{x}$ and a negative reward otherwise:

$$R(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \begin{cases} 1, & \text{if } \hat{\boldsymbol{x}} = \boldsymbol{x} \\ -1, & \text{if otherwise.} \end{cases} \tag{2}$$

Lastly, note that the original LG is a specification of the NLG where we set $\lambda = 0$. Figure 1 depicts a visual representation of NLG. For completeness, the LG is depicted in Appendix B.

2.2 Agent Architectures

We now describe the architectures implemented for both agents, the Speaker and the Listener (Figure 2). We design both architectures to be able to model policy gradient RL algorithms [49].

As an overview, the Speaker's objective is to encode a discrete message, $\boldsymbol{m}$, describing an input image, $\boldsymbol{x}$, see Section 2.1. First, we encode the image using a pre-trained image encoder [15], f, to reduce its dimensionality and extract valuable features, $\boldsymbol{x}' = f(\boldsymbol{x})$. Subsequently, a trainable encoder g processes the new sequence of features, outputting the initial hidden and cell values, $(\boldsymbol{z}_{0,\boldsymbol{\theta}}, \boldsymbol{c}_{0,\boldsymbol{\theta}}) = g(\boldsymbol{x}'; \boldsymbol{\theta})$, used by the recurrent module h, in this case, an LSTM [20].

Subsequently, the Speaker will select each token m_t to add to the message iteratively, using h. On this account, we define a complementary embedding module, e, to convert the previous discrete token m_{t-1} into a dense vector $\boldsymbol{d}_{t,\boldsymbol{\theta}} = e(m_{t-1}; \boldsymbol{\theta})$. Then, the recurrent module, h, consumes the new dense vector and previous internal states to produce the new ones, $(\boldsymbol{z}_{t,\boldsymbol{\theta}}, \boldsymbol{c}_{t,\boldsymbol{\theta}}) = h(\boldsymbol{d}_{t,\boldsymbol{\theta}}, \boldsymbol{z}_{t-1,\boldsymbol{\theta}}, \boldsymbol{c}_{t-1,\boldsymbol{\theta}}; \boldsymbol{\theta})$. We then pass $\boldsymbol{z}_{t,\boldsymbol{\theta}}$ through two concurrent heads: (i) The actor head yields the probability of choosing each token as the next one, $m_t \sim \pi_S(\cdot|\boldsymbol{z}_{t,\boldsymbol{\theta}}; \boldsymbol{\theta})$; (ii) The critic head estimates the expected reward $V(\boldsymbol{x}) := v(\boldsymbol{z}_{t,\boldsymbol{\theta}}; \theta)$. After the new token is sampled, we feed it back to $e(\cdot; \theta)$, and the process repeats itself until we generate N tokens. The first token m_0 is a predefined *start-of-string* token and is not included in the message. Following [6], we maintain the original vocabulary and message sizes, where $|\mathbb{W}| = 20$, and $N = 10$, making the set of all possible message much larger than the size of the dataset used ($|\mathbb{X}| \approx 10^6$ for ImageNet [45]). We depict the Speaker's architecture in Figure 2a.

The Listener architecture has two different modules to process the message, sent from the Speaker, $\boldsymbol{m}$ and the images obtained as candidates $\mathbb{C}$. Additionally, a third module combines the output of both input components and provides it to the actor and critic heads. We now describe each component.

To process the candidate images $\mathbb{C}$, the Listener uses the same pre-trained encoder f combined with the network c to embed the candidate images, $\boldsymbol{l}_j = c(\boldsymbol{x}'_j; \boldsymbol{\phi})$, where $\boldsymbol{x}'_j = f(\boldsymbol{x}_j)$ and $\boldsymbol{x}_j \in \mathbb{C}$.

Concerning the message received, the Listener uses the recurrent model h (an LSTM) to handle each token, m_t, iteratively. Similarly to the Speaker, there is an embedding layer, $e(\cdot; \boldsymbol{\phi})$, to convert the discrete token into a dense vector before giving it to h, where we have $(\boldsymbol{z}_{t,\boldsymbol{\phi}}, \boldsymbol{c}_{t,\boldsymbol{\phi}}) = h(e(m_t; \boldsymbol{\phi}), \boldsymbol{z}_{t-1,\boldsymbol{\phi}}, \boldsymbol{c}_{t-1}; \boldsymbol{\phi})$. The initial internal states of h are initialized as $\boldsymbol{z}_{0,\boldsymbol{\phi}} = \mathbf{0}$ and $\boldsymbol{c}_{0,\boldsymbol{\phi}} = \mathbf{0}$. After processing all message tokens, the final hidden state, $\boldsymbol{z}_{N,\boldsymbol{\phi}}$, goes through a final network g to output the message's hidden value $\boldsymbol{l}_{\text{m}} = g(\boldsymbol{z}_{N,\boldsymbol{\phi}}; \boldsymbol{\phi})$. Finally, the generated hidden values for the message and all candidates flow through to the head module.

The first operation in the head module executes an attention mechanism to combine the message features with each candidate's counterpart. The output includes a value per candidate which we concatenate into a vector $\boldsymbol{s} = \begin{bmatrix} \boldsymbol{l}_{\text{m}} \cdot \boldsymbol{l}_1 & \dots & \boldsymbol{l}_{\text{m}} \cdot \boldsymbol{l}_{|\mathbb{C}|} \end{bmatrix}^T$, called the candidates' score. We define the actor head as $\pi_L(\cdot|\boldsymbol{s}; \boldsymbol{\phi})$ to output the Listener's policy $\hat{\boldsymbol{x}} \sim \pi_L(\cdot|\boldsymbol{m}, \mathbb{C})$, which is a valid approximation since $\boldsymbol{s}$ holds information from the message and candidates. Parallelly, the critic head $v(\cdot; \boldsymbol{\phi})$ receives the same scores $\boldsymbol{s}$ and estimates the expected cumulative reward, as detailed in Section 2.3.

2.3 Learning Strategy

As described at the start of Section 2.1, the agents can only transmit information via the communication channel, which has only one direction: from the Speaker to the Listener. Additionally, agents learn how to communicate following the RIAL protocol, where agents are independent and treat others as part of the environment. As such, we have a decentralized training scheme where the agents improve their own parameters solely by maximizing the game's reward, see equation 2.

To perform well and consistently when playing the NLG, the Speaker must learn how to utilize the vocabulary to distinctively encode each image into a message to obtain the highest expected reward possible. We use Reinforce [54], a policy gradient algorithm, to train the Speaker. Given a target image $\boldsymbol{x}$ and the corresponding Listener's action $\hat{\boldsymbol{x}}$, we have a loss, $L_{\text{S,A}}$, to fit the actor's head and another one, $L_{\text{S,C}}$, for the critic's head. We define,

$$L_{\text{S,A}}(\boldsymbol{\theta}) = -\sum_{t=1}^{N} \text{sg}\left(R(\boldsymbol{x}, \hat{\boldsymbol{x}}) - v\left(\boldsymbol{z}_{t,\boldsymbol{\theta}}; \boldsymbol{\theta}\right)\right) \cdot \log \pi_S\left(m_t | \boldsymbol{z}_{t,\boldsymbol{\theta}}; \boldsymbol{\theta}\right),$$

where $\text{sg}(\cdot)$ is the *stop-gradient* function, in order to optimize the policy. Note that the Speaker is in a sparse reward setting [39], where the sum of returns is the same as the game reward $R(\boldsymbol{x}, \hat{\boldsymbol{x}})$. Further, we subtract a baseline (critic head's value $v\left(\boldsymbol{z}_{t,\boldsymbol{\theta}}\right)$) from

Figure 2: Graphical representation of Speaker, Figure 2a, and Listener, Figure 2b, architectures for the NLG. In this illustration, the message, m, contains only two tokens, $N = 2$.

the returns to reduce variance. Regarding the critic loss, we devise

$$L_{\text{S,C}}(\boldsymbol{\theta}) = \sum_{t=1}^{N} \left(R(\boldsymbol{x}, \hat{\boldsymbol{x}}) - v(\boldsymbol{z}_{t,\boldsymbol{\theta}}; \boldsymbol{\theta})\right)^2,$$

to approximate the state-value function $V(\boldsymbol{x}) = \mathbb{E}_{\pi_S}\left[R(\boldsymbol{x}, \hat{\boldsymbol{x}})\right]$.

We also use an additional entropy regularization term, $L_{\text{S},\mathcal{H}}$, to make sure the language learned by the Speaker will not entirely stagnate by encouraging new combinations of tokens that increase entropy, further incentivizing exploration. Moreover, we define a target policy for the Speaker to minimize an additional KL divergent term, $L_{\text{S,KL}}$, between the online and target policies, $\boldsymbol{\theta}$ and $\bar{\boldsymbol{\theta}}$, respectively. We update $\bar{\boldsymbol{\theta}}$ using an exponential moving average (EMA) over $\boldsymbol{\theta}$, *i.e.* $\bar{\boldsymbol{\theta}} \leftarrow (1-\eta)\boldsymbol{\theta} + \eta\bar{\boldsymbol{\theta}}$ where η is the EMA weight parameter. With $L_{\text{S,KL}}$, we prevent steep changes in the parameter space, which helps stabilize training [7, 42]. We refer to Chaabouni et al. [6] for a complete analysis on the impact of $L_{\text{S,KL}}$. Finally, we weigh each loss term and average the resulting sum given a batch of input images, $\mathbb{X}' \subset \mathbb{X}$, to obtain the overall Speaker loss:

$$L_{\text{S}}(\boldsymbol{\theta}) = \frac{1}{|\mathbb{X}'|} \sum_{\boldsymbol{x} \in \mathbb{X}'} \alpha_{\text{S,A}} L_{\text{S,A}}(\boldsymbol{\theta}) + \alpha_{\text{S,C}} L_{\text{S,C}}(\boldsymbol{\theta}) + \alpha_{\text{S},\mathcal{H}} L_{\text{S},\mathcal{H}}(\boldsymbol{\theta}) + \alpha_{\text{S,KL}} L_{\text{S,KL}}(\boldsymbol{\theta}),$$

where $\alpha_{\text{S,A}}$, $\alpha_{\text{S,C}}$, $\alpha_{\text{S},\mathcal{H}}$, $\alpha_{\text{S,KL}}$ are constants.

We also use Reinforce [54] to train the Listener. We define the loss $L_{\text{L,A}}$ to train the Listener's policy:

$$L_{\text{L,A}}(\boldsymbol{\phi}) = -\,\text{sg}\left(R(\boldsymbol{x}, \hat{\boldsymbol{x}}) - v(\boldsymbol{s}; \boldsymbol{\phi})\right) \cdot \log \pi_L(\hat{\boldsymbol{x}} | \boldsymbol{s}; \boldsymbol{\phi}),$$

where cumulative returns is again the game reward $R(\boldsymbol{x}, \hat{\boldsymbol{x}})$ since the Listener is in a single-step episode format where the game ends after choosing a candidate, $\hat{\boldsymbol{x}} \in \mathbb{C}$. Identically to the Speaker, we subtract the Listener critic's value $v(s; \boldsymbol{\phi})$ from the game reward. The critic sub-network optimizes

$$L_{\text{L,C}}(\boldsymbol{\phi}) = \left(R(\boldsymbol{x}, \hat{\boldsymbol{x}}) - v(s; \boldsymbol{\phi})\right)^2.$$

Similarly to the Speaker loss, we add an entropy loss term $L_{\text{L},\mathcal{H}}(\boldsymbol{\phi})$ to encourage exploration. The final Listener loss for a batch of images $\mathbb{X}'$ is:

$$L_{\text{L}}(\boldsymbol{\phi}) = \frac{1}{|\mathbb{X}'|} \sum_{x \in \mathbb{X}'} \alpha_{\text{L,A}} L_{\text{L,A}}(\boldsymbol{\phi}) + \alpha_{\text{L,C}} L_{\text{L,C}}(\boldsymbol{\phi}) + \alpha_{\text{L},\mathcal{H}} L_{\text{L},\mathcal{H}}(\boldsymbol{\phi}),$$

where $\alpha_{\text{L,A}}$, $\alpha_{\text{L,C}}$, and $\alpha_{\text{L},\mathcal{H}}$ are constants.

A detailed analysis of the learning strategy, for both agents, can be found in Appendix E.1. Additionally, due to the complexity and non-stationarity of NLG, we define a scheduler for the noise level in the communication channel, during training. Namely, we linearly increase the noise level from 0 to λ at the beginning of training. This phase is optional and only helps with data efficiency (we refer to Appendix E.4 for more details).

3 EVALUATION

We provide an extensive evaluation of NLG and variants. For completeness, we also consider the original architecture proposed by Chaabouni et al. [6] and our novel agent architecure to play the original LG (without message noise) as baselines. In this game variant, our model surpasses the original architecture at a slight cost of data efficiency. This trade-off is expected and fully explained in Section 3.2. At a glance, this happens because the baseline version can retrieve more information than our implementation, during training. Having a progressive sequence of LG variants enables us to assess how each modification influences the emergent communication protocol learned by the agents.

We continue this section by introducing all LG variants, giving a broader view of each game, agent architectures, and learning strategy. Next, we evaluate the generality of the emerging language for each game variant when providing new and unseen images. We compare LG and NLG variants when testing with and without noise in the communication channel. Additionally, we investigate how the candidate set, $\mathbb{C}$, impacts the generalization capabilities of the message protocols. Moreover, we investigate the internal message structure to understand how robust communication protocols emerge when agents play in the NLG. Finally, we perform an out-of-distribution evaluation to discern how each variant reacts to novel forms of noise. We always report results using the average (plus SD) over 10 different seeds.

Due to space constraints, Appendix F contains the results obtained in all experiments, for both ImageNet [45] and CelebA [32] datastes, used to devise the analyses detailed in this section. Additionally, we report a supplementary evaluation to assess the capacity of each game variant to adapt to new tasks in a transfer learning manner, called *ease and transfer learning* (ETL) [6] (see Appendix G). This supplementary evaluation gives yet another

frame of reference to evaluate the generality and robustness of the learned languages. Finally, we also refer to the appendix for further details regarding related work (Appendix A), every game variant (Appendix B), model architectures (Appendix D), and datasets used (Appendix E.2).

3.1 Lewis Game Variants

We briefly report essential aspects of each game variant, while referring to supplementary information when necessary. We consider three variants of the LG, all of which share the same Speaker architecture. The Listener architecture differs in all games. We refer to Appendix D for a detailed description of the implementation of these architectures. Additionally, all variants except for LG (S) are a contribution of this work. The LG variants considered are:

- LG (S): Original LG variant introduced in Chaabouni et al. [6]. The Listener trains under supervised data by using InfoNCE loss [11, 37] to find similarities between the message and the correct candidate, see Appendix C.
- LG (RL): LG with a deterministic communication channel (no noise) where both agents implement RL architectures. The only semi-supervised information given to both agents is whether the game ended successfully or not. We refer to Sections 2.2 and 2.3 for a comprehensive description of the agents' architectures and learning strategies, respectively.
- NLG: LG variant introduced in Section 2.1, where we apply an external environmental pressure by adding noise to the message during transmission. Both agents function as RL agents, as in LG (RL). Agents' architectures and learning strategies appear in Sections 2.2 and 2.3, respectively. For an overall understanding of NLG, we define 3 different versions: NLG (0.25), NLG (0.5), and NLG (0.75); where NLG (x) means that, during training, we fix the noise threshold at $\lambda = x$.

3.2 Robust Communication Protocols

This section analyzes the performance of all LG variants described above. Since there is the possibility to apply different hyper-parameters depending on the current phase (training or testing phase), we define two extra variables, λ_{test} and $\mathbb{C}_{\text{test}}$, to define the noise threshold and candidate set applied during the test phase, respectively.

Starting by comparing LG (S) with LG (RL), we can see an apparent performance boost for the LG when the Listener is an RL agent. Figure 3 shows that, during training, the RL version performs better than the supervised version. Equivalent results occur in the testing phase. From Figures 4 and 5, and focusing on the results obtained with a deterministic communication channel, $\lambda_{\text{test}} = 0$, the RL version surpasses the accuracy achieved by the supervised counterpart. This performance gap becomes more predominant as we increase game complexity, as seen in Section 3.2.2. From Figure 3, we also observe a trade-off between performance and sample efficiency, where the RL version is less sample efficient. We can trace these differences back to the loss function employed by each version. For instance, the supervised version employs the InfoNCE loss (Appendix D.2.2), which we can see as a Reinforce variant with only a policy to optimize and, particularly, with access to an oracle giving information about which action (candidate) is the right one for each received message. As such, the Listener (S) can efficiently learn how to map messages to the correct candidates. On the other hand, the RL version has no access to such oracle and needs to interact with the environment to build this knowledge. The decrease in sample efficiency from supervised to RL is, therefore, a natural phenomenon. Nonetheless, the RL version introduces a critic loss term whose synergy with the policy loss term helps to improve the final performance when compared to the supervised version.

One disadvantage of employing, at inference time, the communication protocols learned by playing default LG variants (LG (S) and LG (RL)) is that they are not robust to deal with message perturbations. Since agents train only with perfect communication, they never experience noisy communication. When testing the performance of LG (S) and LG (RL) with noisy communication channels $\lambda_{\text{test}} > 0$, we observe a noticeable dominance of RL against S. Nonetheless, there is a massive drop in performance for both variants compared to the noiseless case $\lambda_{\text{test}} = 0$, see Figures 4 and 5.

Conversely, NLG puts agents in a more complex environment where only random fractions of the message are visible during training time. Despite such modifications, the pair of agents can still adapt to the environment and learn robust communication protocols that handle both types of messages (with and without noise). We notice equivalent accuracy performance for NLG and LG (RL) when testing with deterministic communication channels, see Figures 4 and 5. Notably, every NLG version only suffers a negligible performance loss when testing with $\lambda_{\text{test}} = 0.25$. This loss starts to be more noticeable at higher noise levels, where the accuracy drops to around 80%, and further to the interval between 30%-40% when λ_{test} is 0.5 and 0.75, respectively. Still, NLG is considerably more effective than LG (S) and LG (RL) when communicating in noisy environments, as seen in the considerable performance gap visible in each tested noise level λ_{test}. This increased performance suggests that, in NLG, agents can encode redundant information where communication is still functional when random parts of the message are hidden. Additionally, the performance obtained for each test threshold λ_{test} is similar for every NLG version. As such, each version displays similar capacities to handle noise in the communication channel, independent of the noise threshold λ applied during training. Please refer to Appendix F for additional results on the ImageNet and CelebA datasets.

3.2.1 Comparing Different Noise Levels. Comparing the different variants of NLG, we observe that the mean accuracy obtained for NLG (0.75) is slightly lower than NLG (0.25) and NLG (0.5) when we set λ_{test} to 0 or 0.25, see Figures 4 and 5. When $\lambda_{\text{test}} = 0.5$, NLG (0.5) performs slightly better than its counterparts. Finally, NLG (0.5) and NLG (0.75) seem to perform slightly better than NLG (0.25) is the extreme noise case (λ_{test} is 0.75).

Henceforth, having a threshold of $\lambda = 0.5$ during training appears to give a good balance for the pair of agents to develop a communication protocol that can effectively act in a broad range of noise levels, even when there is no noise during communication.

3.2.2 Scaling the Number of Candidates. We train every game variant with different candidate sizes $|\mathbb{C}|$. We scale $|\mathbb{C}|$ from 16 to 1024, using a ratio of 4. At test time, we evaluate all experiments using larger sizes, for the candidates set, to inspect generalization capabilities. In our case, we use $|\mathbb{C}_{\text{test}}| = 1024$ and $|\mathbb{C}_{\text{test}}| = 4096$. Looking

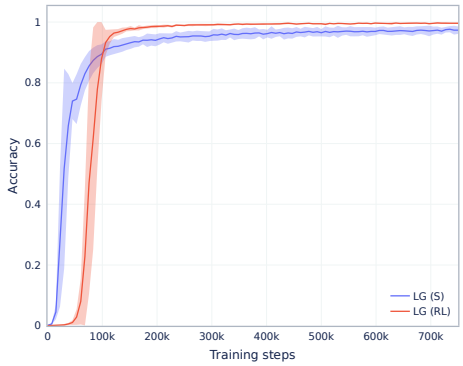

Figure 3: Training accuracy for LG (S) and LG (RL). Trained on ImageNet dataset and $|\mathbb{C}| = 1024$.

at Tables 1 and 2, we can see an evident generalization boost when the number of candidates increases for every game. We posit that increasing the game's difficulty (increasing the number of candidates) helps the agents to generalize. As the candidates' set gets additional images, the input diversity increases, which affects how agents encode and interpret more information to distinguish the correct image from all others. Even when adding noise, the agents can quickly adapt to such changes conveying information in such a way as to repress the noise mechanism. We argue that agents have two correlated ways to achieve such adaptation: (1) send redundant information regarding specific features of the image; and (2) create a spatial mapping from the image to the message space.

We note that as $|\mathbb{C}|$ increases, the test performance also increases, but at a smaller scale. As an example, consider LG (RL) variant when $\lambda_{\text{test}} = 0$ and $|\mathbb{C}_{\text{test}}| = 4096$. In this case, the test performance gap between $|\mathbb{C}| = 16$ and $|\mathbb{C}| = 64$ is 0.4, and only 0.03 between $|\mathbb{C}| = 256$ and $|\mathbb{C}| = 1024$, see Table 1. Regarding NLG, the accuracy starts lower for smaller candidate sizes, e.g., 0.27 when $|\mathbb{C}| = 16$, against 0.67 for the LG (RL) counterpart. Nonetheless, as the candidate set size increases, the noise effect becomes less predominant and the NLG's performance reaches the same level as in LG (RL), both achieving an accuracy of 0.97 when $|\mathbb{C}| = 1024$.

Equivalently to the results introduced in the beginning of Section 3.2, we observe low accuracy for LG (RL) when testing with noisy communication channels (Table 2), regardless of the candidate size. In respect to NLG, there is an apparent increase in performance as $|\mathbb{C}|$ increases during training. Additionally, the performance gap between consecutive candidate set sizes is approximately the same (between 0.14 and 0.21, when $|\mathbb{C}_{\text{test}}| = 4096$).

3.3 Message Structure Analysis

With the aim of addressing how NLG variations develop robust communication protocols, we propose to analyze the message structure of the language protocols. The NLG includes additional adverse pressures where the communication channel seems unreliable. In order to overcome the noise introduced by the faulty channel, the pair of agents must find alternative ways to coordinate how to send information. Moreover, since the only feedback received by the Speaker is the game's outcome, the new coordination mechanism becomes essential to complete the game with a high success rate. The most intuitive behavior for the Speaker focuses on developing implicit repair mechanisms where messages incorporate redundant information. As such, even if a subset of the message tokens become masked, the Listener can still parse the remaining message and select the correct candidate, making the communication robust to the noise being introduced by the faulty channel.

To test our assumption, we indirectly analyze the internal structure of the message being transmitted in all games. Since we want to examine the existence of redundancy in the communication protocol, we propose an evaluation where we iteratively increase the number of masked tokens and retrieve the performance obtained. The range of masked tokens spreads from 0 to $N/2 = 5$ (half of the tokens). Additionally, when selecting the number of tokens to conceal, we evaluate the performance over ten different combinations of masked tokens (except when no tokens are masked), ensuring the results represent the average case. As such, this analysis allows us to examine, in greater detail, how the accuracy changes as we increase the number of masked tokens iteratively. As shown in Figures 6 and 7, the NLG variations only decrease slightly in accuracy as the number of masked tokens increases, conveying that our assumption is accurate and messages contain redundant information. Consequently, the Listener still contains enough information to select the right candidate, even with partial messages. On the other hand, for the deterministic variations, LG (S) and LG (RL), we observe a faster decrease in performance as more tokens are masked, conveying that every token is essential for the Listener to infer the right candidate. Furthermore, we highlight that the average accuracy obtained by the NLG (0.5) variant, with five tokens masked, is similar to the performance obtained by LG (RL) when masking a single token, which is around 75% (Figure 7). This particular result illustrates how much noise both variants can manage for a particular and similar performance level. As such, NLG (0.5) can deal with 5x more noise than LG (RL) when discriminating the most amount of images, $|\mathbb{C}_{\text{test}}| = 4096$. Please refer to Appendix F for additional results on the ImageNet and CelebA datasets.

In addition to the results presented above, we observe an interesting occurrence when communication protocols emerge with deterministic channels, as in the case of LG (S) and LG (RL). Our analysis shows that when masking a single token, the first token of the message seems to carry more information than the others, or, at least, is crucial to derive meaning from the rest of the message. We show these results in the Appendix F, where the drop in performance can, in some cases, drop down around 10% when the first token is masked in opposition to mask any other token. The drop in performance is more noticeable as the number of candidates given to the Listener increases, where it seems the first token plays a crucial role to reduce the number of final choices to consider.

3.4 External Noise Interference

In this experiment, we test the capability of the emergent communication protocols trained in the games presented above to adapt to new game variations where we focus on adding noise to conceal information in other components of the input. In this regard, we model a new noise dynamic by adding random information to the objects to discriminate: (1) the target image provided to the Speaker, and (2) the candidates given to the Listener. We sample noise from

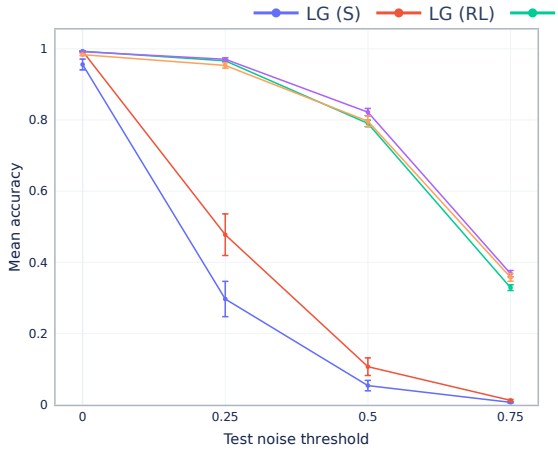

Figure 4: Test accuracy for all variants with $|\mathbb{C}_{\text{test}}| = 1024$. Trained on ImageNet and $|\mathbb{C}| = 1024$. The x-axis denotes λ_{test}.

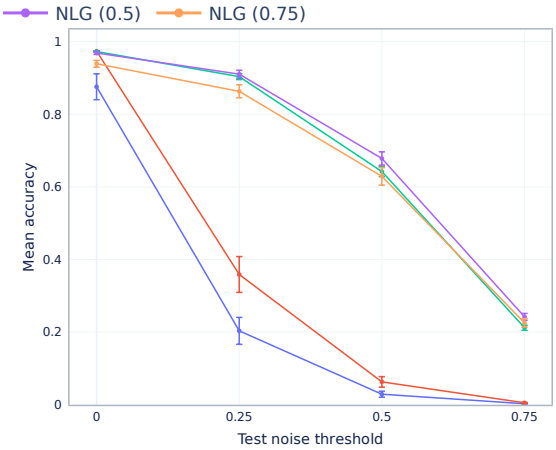

Figure 5: Test accuracy for all variants with $|\mathbb{C}_{\text{test}}| = 4096$. Trained on ImageNet and $|\mathbb{C}| = 1024$. The x-axis denotes λ_{test}.

Table 1: Test accuracy including SD, when $\lambda_{\text{test}} = 0$, for LG (RL) and NLG, on ImageNet dataset.

Game	λ	$\lvert\mathbb{C}\rvert$	$\lvert\mathbb{C}\rvert$ (test)	
			1024	4096
LG (RL)	0	16	0.67 (0.04)	0.39 (0.04)
LG (RL)	0	64	0.93 (0.01)	0.79 (0.03)
LG (RL)	0	256	0.98 (0.00)	0.94 (0.01)
LG (RL)	0	1024	0.99 (0.00)	0.97 (0.00)
NLG	0.5	16	0.55 (0.03)	0.27 (0.02)
NLG	0.5	64	0.87 (0.01)	0.67 (0.03)
NLG	0.5	256	0.98 (0.00)	0.91 (0.01)
NLG	0.5	1024	0.99 (0.00)	0.97 (0.00)

Table 2: Test accuracy including SD, when $\lambda_{\text{test}} = 0.5$, for LG (RL) and NLG, on ImageNet dataset.

Game	λ	$\lvert\mathbb{C}\rvert$	$\lvert\mathbb{C}\rvert$ (test)	
			1024	4096
LG (RL)	0	16	0.03 (0.01)	0.01 (0.01)
LG (RL)	0	64	0.09 (0.07)	0.05 (0.03)
LG (RL)	0	256	0.11 (0.06)	0.06 (0.04)
LG (RL)	0	1024	0.11 (0.02)	0.06 (0.01)
NLG	0.5	16	0.32 (0.02)	0.14 (0.01)
NLG	0.5	64	0.57 (0.02)	0.33 (0.02)
NLG	0.5	256	0.73 (0.01)	0.54 (0.02)
NLG	0.5	1024	0.82 (0.01)	0.68 (0.02)

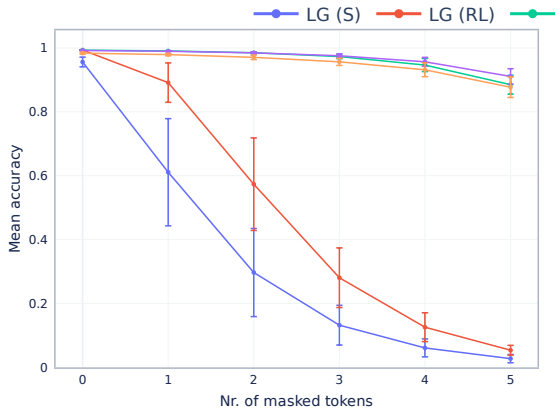

Figure 6: Mean test accuracy for all LG variants, trained with $|\mathbb{C}| = 1024$ and on ImageNet dataset. At test time, $|\mathbb{C}_{\text{test}}| = 1024$. We report 10 different combinations of masked tokens.

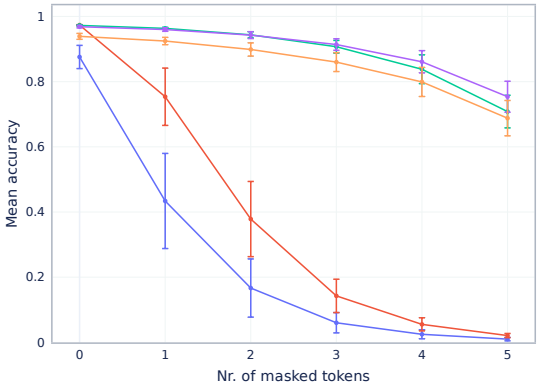

Figure 7: Mean test accuracy for all LG variants, trained with $|\mathbb{C}| = 1024$ and on ImageNet dataset. At test time, $|\mathbb{C}_{\text{test}}| = 4096$. We report 10 different combinations of masked tokens.

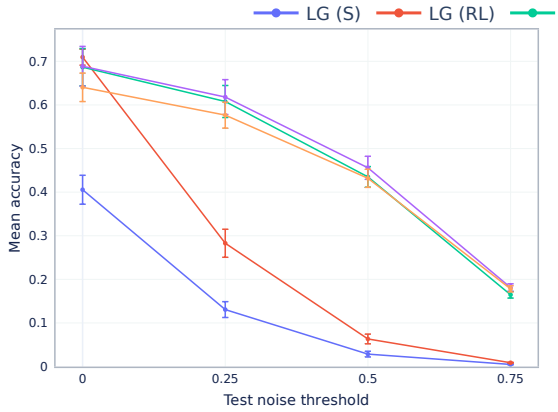

Figure 8: Mean test accuracy for all variants with $|\mathbb{C}_{\text{test}}| = 1024$. Trained on ImageNet and $|\mathbb{C}| = 1024$. During test noise is added to the inputs (Section 3.4). The x-axis denotes λ_{test}.

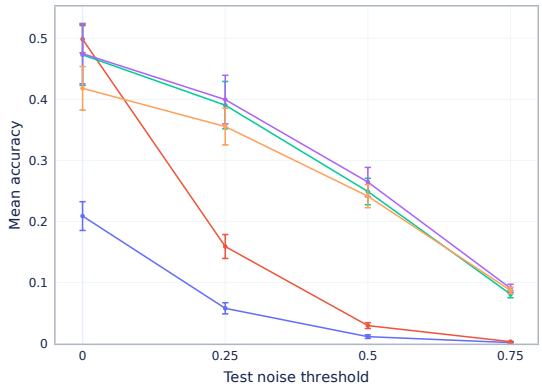

Figure 9: Mean test accuracy for all variants with $|\mathbb{C}_{\text{test}}| = 4096$. Trained on ImageNet and $|\mathbb{C}| = 1024$. During test noise is added to the inputs (Section 3.4). The x-axis denotes λ_{test}.

a Gaussian distribution and add it to the output of the frozen image encoder. As such, in this game adaptation, we modify the target image to $\boldsymbol{x}' = f(\boldsymbol{x}) + \boldsymbol{\varepsilon}$ and the candidates to $\boldsymbol{x}_i = f(\boldsymbol{x}_i) + \boldsymbol{\varepsilon}$, where $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \boldsymbol{I})$ and $\boldsymbol{x}_i \in \mathbb{C}$. Unless otherwise noted, we set $\sigma = 1$.

We can view this new modification as an out-of-distribution task since, during training, agents only deal with noise in the communication channel, where the obstruction happens by masking some of the message tokens. Figures 8 and 9 depict the results obtained for all game variants at inference time when we introduce noise to the input images. When testing with a deterministic channel ($\lambda_{\text{test}} = 0$), LG (RL) and all NLG versions have similar performance, where the mean accuracy is around 0.7 and 0.5 for 1024 and 4096 candidates, respectively. We can see a slight degradation in performance across all variants compared to the previous experiment setting in Section 3.2 (images without any perturbations). Nonetheless, these results indicate a positive transfer capability where the language protocols are general enough, allowing effective agent reasoning even with partially hidden input distributions. Additionally, we notice that adding noise in the communication channel during training (NLG variant) does not provide improved benefits to the communication protocol to deal with other noise types, as there is no gain in performance when $\lambda_{\text{test}} = 0$, comparing against LG (RL).

For the experiments where noise is also present in the communication channel, $\lambda_{\text{test}} > 0$, we observe similar results as in Section 3.2, where NLG versions vastly surpass both LG (RL) and LG (S) since the former emergent protocols can efficiently retrieve applicable information to reason about, from the noisy messages. Finally, the results obtained by LG (S) were considerably lower than all other variants, again claiming the superiority of having a Listener as an RL agent, see Section 3.2. Please refer to Appendix F for additional results on the ImageNet and CelebA datasets.

4 CONCLUSION & FUTURE WORK

In this work, we focus on designing agent systems that can learn language protocols without prior knowledge, where communication evolves grounded on experience to solve the task at hand. We explore EC from a language evolution perspective to analyze a particular linguistic concept called conversational repair. Conversational repair appears in human languages as a mechanism to detect and resolve miscommunication and misinformation during social interactions. Mainly, we focus on the implicit repair mechanism, where the interlocutor sending the information deliberately communicates in such a way as to prevent misinformation and avoid future interactions on correcting it. Our analysis shows that implicit conversational repair can also emerge in artificial designs when there is enough disruptive environmental pressure, where sending redundant information facilitates solving the task more effectively.

For future work, several ideas can be explored. One possible research idea passes to merge explicit and implicit repair mechanisms. In this scenario, agents would have to coordinate between using implicit mechanisms to prevent misinformation or starting a posterior dialogue (an explicit mechanism) when the Listener cannot extract useful information from the message sent. We argue that this coordination mechanism can emerge naturally in sufficiently complex environments, where the Listener develops the capacity to leverage the likelihood of success and the cost of playing more communication rounds. Another exciting research direction focuses on developing a universal repair mechanism system with the objective of merging various language protocols, grounded in different tasks, into a universal and general language protocol, focusing on the capability to be used by new agents and in out-of-distribution tasks.

ACKNOWLEDGMENTS

This work was supported by national funds through Fundação para a Ciência e a Tecnologia (FCT) with reference UIDB/50021/2020 - DOI: 10.54499/UIDB/50021/2020, project Center for Responsible AI with reference C628696807-00454142, and project RELEvaNT PTDC/CCI-COM/5060/2021. The first author acknowledges the FCT PhD grant 2022.14163.BD.

REFERENCES

[1] Saul Albert and Jan P De Ruiter. 2018. Repair: the interface between interaction and cognition. *Topics in cognitive science* 10, 2 (2018), 279–313.

[2] Yonatan Bisk, Ari Holtzman, Jesse Thomason, Jacob Andreas, Yoshua Bengio, Joyce Chai, Mirella Lapata, Angeliki Lazaridou, Jonathan May, Aleksandr Nisnevich, et al. 2020. Experience grounds language. *arXiv preprint arXiv:2004.10151* (2020).

[3] Ben Bogin, Mor Geva, and Jonathan Berant. 2018. Emergence of communication in an interactive world with consistent speakers. *arXiv preprint arXiv:1809.00549* (2018).

[4] Rahma Chaabouni, Eugene Kharitonov, Diane Bouchacourt, Emmanuel Dupoux, and Marco Baroni. 2020. Compositionality and Generalization In Emergent Languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (Eds.). Association for Computational Linguistics, Online, 4427–4442. https://doi.org/10.18653/v1/2020.acl-main.407

[5] Rahma Chaabouni, Eugene Kharitonov, Emmanuel Dupoux, and Marco Baroni. 2019. Anti-efficient encoding in emergent communication. *Advances in Neural Information Processing Systems* 32 (2019).

[6] Rahma Chaabouni, Florian Strub, Florent Altché, Eugene Tarassov, Corentin Tallec, Elnaz Davoodi, Kory Wallace Mathewson, Olivier Tieleman, Angeliki Lazaridou, and Bilal Piot. 2022. Emergent Communication at Scale. In *International Conference on Learning Representations*. https://openreview.net/forum?id=AUGBfDIV9rL

[7] Elliot Chane-Sane, Cordelia Schmid, and Ivan Laptev. 2021. Goal-conditioned reinforcement learning with imagined subgoals. In *International Conference on Machine Learning*. PMLR, 1430–1440.

[8] Colin Cherry. 1966. On human communication. (1966).

[9] Edward Choi, Angeliki Lazaridou, and Nando de Freitas. 2018. Compositional Obverter Communication Learning from Raw Visual Input. In *International Conference on Learning Representations*.

[10] Tri Dao and Albert Gu. 2024. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. *arXiv preprint arXiv:2405.21060* (2024).

[11] Roberto Dessi, Eugene Kharitonov, and Baroni Marco. 2021. Interpretable agent communication from scratch (with a generic visual processor emerging on the side). In *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (Eds.), Vol. 34. Curran Associates, Inc., 26937–26949. https://proceedings.neurips.cc/paper_files/paper/2021/file/e250c59336b505ed411d455abaa30b4d-Paper.pdf

[12] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783* (2024).

[13] Jakob Foerster, Ioannis Alexandros Assael, Nando De Freitas, and Shimon Whiteson. 2016. Learning to communicate with deep multi-agent reinforcement learning. *Advances in neural information processing systems* 29 (2016).

[14] Laura Graesser, Kyunghyun Cho, and Douwe Kiela. 2019. Emergent linguistic phenomena in multi-agent communication games. In *2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019*. Association for Computational Linguistics, 3700–3710.

[15] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. 2020. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems* 33 (2020), 21271–21284.

[16] Shangmin Guo, Yi Ren, Serhii Havrylov, Stella Frank, Ivan Titov, and Kenny Smith. 2019. The emergence of compositional languages for numeric concepts through iterated learning in neural agents. *arXiv preprint arXiv:1910.05291* (2019).

[17] Serhii Havrylov and Ivan Titov. 2017. Emergence of language with multi-agent games: Learning to communicate with sequences of symbols. *Advances in neural information processing systems* 30 (2017).

[18] Makoto Hayashi, Geoffrey Raymond, and Jack Sidnell. 2013. *Conversational repair and human understanding*. Number 30. Cambridge University Press.

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[20] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[21] Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144* (2016).

[22] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7B. *arXiv preprint arXiv:2310.06825* (2023).

[23] Emilio Jorge, Mikael Kågebäck, Fredrik D Johansson, and Emil Gustavsson. 2016. Learning to play guess who? and inventing a grounded language as a consequence. *arXiv preprint arXiv:1611.03218* (2016).

[24] Vijay Konda and John Tsitsiklis. 1999. Actor-Critic Algorithms. In *Advances in Neural Information Processing Systems*, S. Solla, T. Leen, and K. Müller (Eds.), Vol. 12. MIT Press. https://proceedings.neurips.cc/paper_files/paper/1999/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf

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

[26] Angeliki Lazaridou, Karl Moritz Hermann, Karl Tuyls, and Stephen Clark. 2018. Emergence of Linguistic Communication from Referential Games with Symbolic and Pixel Input. In *6th International Conference on Learning Representations, ICLR 2018-Conference Track Proceedings*.

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

[28] Oliver Lemon. 2022. Conversational grounding in emergent communication–data and divergence. In *Emergent Communication Workshop at ICLR 2022*.

[29] David Lewis. 1979. Scorekeeping in a language game. *Journal of Philosophical Logic* 8, 1 (01 Jan 1979), 339–359. https://doi.org/10.1007/BF00258436

[30] Fushan Li and Michael Bowling. 2019. Ease-of-teaching and language structure from emergent communication. *Advances in neural information processing systems* 32 (2019).

[31] Fushan Li and Michael Bowling. 2019. Ease-of-Teaching and Language Structure from Emergent Communication. In *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Eds.), Vol. 32. Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2019/file/b0cf188d74589db9b23d5d277238a929-Paper.pdf

[32] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. 2015. Deep Learning Face Attributes in the Wild. In *Proceedings of International Conference on Computer Vision (ICCV)*.

[33] Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101* (2017).

[34] Igor Mordatch and Pieter Abbeel. 2018. Emergence of grounded compositional language in multi-agent populations. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 32.

[35] Mitja Nikolaus. 2023. Emergent Communication with Conversational Repair. In *The Twelfth International Conference on Learning Representations*.

[36] Martin A Nowak and David C Krakauer. 1999. The evolution of language. *Proceedings of the National Academy of Sciences* 96, 14 (1999), 8028–8033.

[37] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748* (2018).

[38] OpenAI. 2023. GPT-4 Technical Report. arXiv:2303.08774 [cs.CL]

[39] Christopher Painter-Wakefield and Ronald Parr. 2012. Greedy algorithms for sparse reinforcement learning. In *Proceedings of the 29th International Coference on International Conference on Machine Learning* (Edinburgh, Scotland) *(ICML'12)*. Omnipress, Madison, WI, USA, 867–874.

[40] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. In *International conference on machine learning*. Pmlr, 1310–1318.

[41] David Premack and Guy Woodruff. 1978. Does the chimpanzee have a theory of mind? *Behavioral and brain sciences* 1, 4 (1978), 515–526.

[42] Konrad Rawlik, Marc Toussaint, and Sethu Vijayakumar. 2012. On stochastic optimal control and reinforcement learning by approximate inference. *Proceedings of Robotics: Science and Systems VIII* (2012).

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

[44] Mathieu Rita, Florian Strub, Jean-Bastien Grill, Olivier Pietquin, and Emmanuel Dupoux. 2022. On the role of population heterogeneity in emergent communication. In *International Conference on Learning Representations*. https://openreview.net/forum?id=5Qkd7-bZfI

[45] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)* 115, 3 (2015), 211–252. https://doi.org/10.1007/s11263-015-0816-y

[46] Emanuel A Schegloff, Gail Jefferson, and Harvey Sacks. 1977. The preference for self-correction in the organization of repair in conversation. *Language* 53, 2 (1977), 361–382.

[47] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

[48] Sainbayar Sukhbaatar, Rob Fergus, et al. 2016. Learning multiagent communication with backpropagation. *Advances in neural information processing systems* 29 (2016).

[49] Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction*. MIT press.

[50] Mycal Tucker, Huao Li, Siddharth Agrawal, Dana Hughes, Katia Sycara, Michael Lewis, and Julie A Shah. 2021. Emergent discrete communication in semantic

spaces. *Advances in Neural Information Processing Systems* 34 (2021), 10574–10586.

[51] Ryo Ueda and Koki Washio. 2021. On the Relationship between Zipf's Law of Abbreviation and Interfering Noise in Emergent Languages. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: Student Research Workshop*. Association for Computational Linguistics, Online, 60–70. https://doi.org/10.18653/v1/2021.acl-srw.6

[52] Nino Vieillard, Tadashi Kozuno, Bruno Scherrer, Olivier Pietquin, Remi Munos, and Matthieu Geist. 2020. Leverage the Average: an Analysis of KL Regularization in Reinforcement Learning. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 12163–12174. https://proceedings.neurips.cc/paper_files/paper/2020/file/8e2c381d4dd04f1c55093f22c59c3a08-Paper.pdf

[53] Kyle Wagner, James A Reggia, Juan Uriagereka, and Gerald S Wilkinson. 2003. Progress in the simulation of emergent communication and language. *Adaptive Behavior* 11, 1 (2003), 37–69.

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

[55] Terry Winograd. 1972. Understanding natural language. *Cognitive psychology* 3, 1 (1972), 1–191.

[56] George Kingsley Zipf. 2013. *The psycho-biology of language: An introduction to dynamic philology*. Routledge.