So many design choices: Improving and interpreting neural agent communication in signaling games

Timothée Bernard
LLF, Université Paris Cité, France
timothee.bernard@u-paris.fr

Timothee Mickus
Helsinki University, Finland
timothee.mickus@helsinki.fi

Abstract

Emergent language games are experimental protocols designed to model how communication may arise among a group of agents. In this paper, we focus on how to improve performances of neural agents playing a signaling game: a *sender* is exposed to an image and generates a sequence of symbols that is transmitted to a *receiver*, which uses it to distinguish between two images, one that is semantically related to the original image, and one that is not. We consider multiple design choices, such as pretraining the visual components of the agents, introducing regularization terms, how to sample training items from the dataset, and we study how these different choices impact the behavior and performances of the agents. To that end, we introduce a number of automatic metrics to measure the properties of the emergent languages. We find that some implementation choices are always beneficial, and that the information that is conveyed by the agents' messages is shaped not only by the game, but also by the overall design of the agents as well as seemingly unrelated implementation choices.

1 Introduction

Emergent language games are experimental protocols designed to model how communication may arise among a group of agents. For the linguist, they can serve as models of how language might have emerged in humans (Nowak et al., 1999; Kirby, 2002; Kirby et al., 2008); for the AI or NLP scientist, they provide an interesting and challenging test-bed for cooperation and communication across distinct neural agents using symbolic channels (Havrylov and Titov, 2017; Zhang et al., 2021).

Our focus in this paper is on signaling games (Lewis, 1969). More precisely, we adopt a setting in which a *sender* is exposed to some data and produces a message that is transmitted to a *receiver*. The receiver has then to answer a question related to the data that the sender was exposed to. Both agents share the common goal of the receiver answering correctly to the question. This common goal encourages the sender to encode relevant information about the input data in its message and in such a way that the receiver can decode it. In the present paper, we show the sender an image, the *original image*. The receiver is shown a pair of images: a *target image*, which is semantically related to the original image, and one unrelated *distractor*. These images all depict a solid on a uniform background; the shape, the size, the position and the color of this object are the same for the original and the target image, while at least one of these features is different for the distractor. Based on the sender's message, the receiver has to guess which image of the pair is the target. We allow the senders to compose sequences of arbitrary symbols of variable length.

One of the long-term goals of the study of such language games is to understand under which conditions emergent communication protocols display language-like features. In particular, compositionality has been a major concern ever since Hockett (1960) and remains so in today's NLP research landscape (Baroni, 2019). In order to observe complex, structured communication protocols, we need to provide the agents with an environment complex enough for such a characteristic to develop. This adds two requirements on the agents' stimuli: the images we show them will need to be structured, and ought to not be discriminated through low-level features (Bouchacourt and Baroni, 2018).

When designing and experimenting with such a signaling game, a number of design choices are left open—ranging from the exact objective optimized by the agents, to the selection of training examples and to whether agents have prior information about their environment. In this paper, we exhaustively study how different choices often encountered in the relevant literature interact, and which combinations of these, if any, yield the most stable, efficient

Findings of the Association for Computational Linguistics: ACL 2023, pages 8399–8413

communication protocols. In addition, we use training data that theoretically allow the agents to ignore one aspect of the images (e.g., the color of the object shown, or its size), so as to test whether the agents do ignore one feature and how implementation choices impact this behavior. To that end, we define four automatic metrics to probe syntactic and semantic aspects of their communication protocols; we believe them to be useful to future emergent communication studies, as the current agreed upon tool set for studying artificial emergent languages remains fairly narrow. These metrics help us assess what the emergent languages have in common and how they differ. We find that language-like characteristics can be driven by seemingly unrelated factors, and that ensuring the emergence of a reliable communication protocol that generalizes to held-out examples requires a careful consideration of how to implement the language game. The main contributions of this work are thus twofold: we report an exhaustive review of implementation choices, and we provide novel automated metrics to study the semantics of emergent communication protocols.

We provide an overview of related works in Section 2. Dataset and game details are presented in Section 3. We describe our implementation variants in Section 4 and our automatic metrics in Section 5. We discuss our results in Section 6.

2 Related work

The signaling game we study in this paper is derived of Lewis' (1969) work; more specifically, we build upon the neural network formulation of Lazaridou et al. (2018) using a symbolic channel (Sukhbaatar et al., 2016; Havrylov and Titov, 2017; Lazaridou et al., 2017). Other formulations that we leave for future study involve multi-turn communication (Jorge et al., 2016; Evtimova et al., 2018, a.o.), populations and generations of agents (e.g., Kirby et al., 2014; Foerster et al., 2016; Ren et al., 2020; Chaabouni et al., 2022) or non-symbolic communication channels (e.g., Mihai and Hare, 2021).

There is a large prior body of research that investigate how specific implementation choices can impact the characteristics of the emergent communication protocol. For instance, Liang et al. (2020) advocate in favor of competition as an environmental pressure for learning composition by only rewarding the fastest of two teams in a multi-turn signaling game. Rita et al. (2022) mathematically demonstrate that the typical losses used to implement Lewis games can be broken down in a information term and a co-adaptation term, and that limiting overfitting on the latter term experimentally leads to more compositional and generalizable protocols. Mu and Goodman (2021) discuss generalization, and how to induce it by modifying the signaling game to involve sets of targets, rather than unique targets per episode. Patel et al. (2021) study a navigation task to show how to foster interpretability, i.e., communication protocols that are grounded in agents' perceptions of their environment. Rita et al. (2020) discuss how encouraging "laziness" in the sender and "impatience" in the receiver shapes the messages so as to exhibit Zipfian patterns. Chaabouni et al. (2019b) use handcrafted languages to study word-order preferences of LSTM-based agents. Kim and Oh (2021) discuss the importance of dataset size, game difficulty and agent population sizes. Bouchacourt and Baroni (2018) study how the visual components of signaling game agents can undermine the naturalness of their communication. Korbak et al. (2019) propose a specific pretraining regimen to foster compositionality.

Another relevant section of the literature discusses automatic metrics designed to capture specific language-like aspects of the emergent protocol. Chief of these is the meaning-form correlation (a.k.a. topographic similarity) of Brighton and Kirby (2006), which quantifies compositionality by measuring whether changes in form are commensurate with changes in meaning (though other metrics exist, e.g., Andreas, 2019). Chaabouni et al. (2020) argue that this metric does not correlate with generalization capabilities, and that it is thus unsuitable for studying compositionality. Mickus et al. (2020) show how it is impacted by other language-like features. Following these remarks, we focus on novel metrics and defer discussions of topographic similarity to Appendix B.1.

3 Experimental setup

Dataset. We construct a dataset of synthetic images depicting solids on gray backgrounds, using `vpython`.[1] They exhibit a combination of five *features*, each of which have two possible *values*: horizontal position (`left`, `right`), vertical position (`top`, `bottom`), object type (`cube`, `sphere`), object

[1] https://pypi.org/project/vpython/

color (red, blue), object size (small, large). We generate 1000 images for each of the 2^5 possible combinations of feature values (or *categories*).

We divide the dataset in two splits: a *training split* and an *evaluation split*.[2] This partition is performed as follows. First, one category is selected as the *seed category*. Then, *base categories* are the 16 categories that differ from the seed category on exactly 0, 2 or 4 features. *Generalization categories* are the 16 remaining categories, that differ from the seed category on exactly 1, 3 or 5 features. Base category images are then further divided 80%–20% between training and evaluation splits. All generalization category images are assigned to the evaluation split. The training split therefore contains only images from base categories while the evaluation split contains both images from base categories and images from generalization categories.

This partition of categories entails that that during training, all training instances involve image categories that differ by at least two features. Hence, agents may entirely disregard one feature (e.g., color) and still manage to perfectly discriminate all training instances. Only during evaluation are they confronted with pairs of categories that differ by a single feature: namely, when the original image is taken from a base category and the distractor image from a generalization one (or vice versa).

Game & model architecture. All of our models are comprised of two agents: a *sender* and a *receiver*. They are trained to solve a *Lewis signaling game* with a single communication turn. The sender is first shown an image I and produces a message: a sequence of up to 10 symbols from an alphabet of size 16. The receiver is then provided as input a target image I' of the same category as I, a distractor image J of a different category, and the message, and has to identify I' as the intended target. This game is illustrated in Figure 1. The original image I differs from the target image I' so as to deter the sender from describing low-level features of the images (e.g., specific pixel brightness, Bouchacourt and Baroni, 2018).

Both agents contain an image encoder, implemented as a convolution stack, and an LSTM to process symbols. The sender's LSTM is primed with the encoded original image representation, and then generates the message. The receiver uses its LSTM to convert the message into a vector; it then computes the dot product between the message encoding and each of the target and distractor images encoding; we infer a probability distribution over the image pair using a softmax function.

[2]We do one such split per model trained.

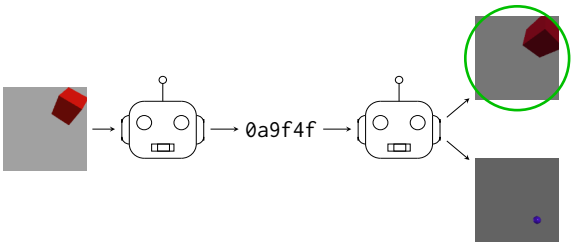

Figure 1: The Lewis Signaling game considered in this paper. The sender (left) is shown the *original image* and produces a message that the receiver (right) uses to distinguish the *target image* from the *distractor image*. The original and target images share the same semantic category (here: `top right big red cube`).

Models are trained with REINFORCE (Williams, 1992); the loss for an episode is defined as:

$$\mathcal{L} = -\sum_t r_t \cdot \log p(a_t) \quad (1)$$

where a_t is the t^{th} action taken in the episode, $p(a_t)$ its probability, and r_t its associated reward. Each episode contains one *generation* action per symbol in the message, and one *classification* action. All actions of an episode are associated with the same reward $r_t = r$. By default, we set r to 1 when the receiver successfully retrieves the target image, and 0 otherwise.

4 Implementation choices

Having described our basic setup above, we now list the different implementation variants that we study in the present paper. We refer to these implementation variants using a vector notation; for a binary trait Φ, a model for which Φ is implemented will be denoted as $\langle \ldots, +\Phi, \ldots, \rangle$, conversely, its absence would be signaled with $\langle \ldots, -\Phi, \ldots, \rangle$.

Pretraining of the visual component. In order to ensure that the recurrent message encoders and decoders receive coherent, usable representations of the images, for some variants, we *pretrain* the image encoders convolutions. In the remainder of the text, we denote as $\langle +P, \ldots \rangle$ models that have undergone pretraining, and $\langle -P, \ldots \rangle$ models that did not. We consider three pretraining objectives: an auto-encoding task and two classification tasks.

The *auto-encoding* pretraining consists in training the convolution stack along with an additional deconvolution stack to reproduce images provided as input, using a mean squared error loss:

$$\mathcal{L}_{\text{AE}} = \frac{1}{3hw} \sum_{i=1}^{h} \sum_{j=1}^{w} \sum_{c=1}^{3} \left(\mathbf{Y}_{ijc} - \hat{\mathbf{Y}}_{ijc} \right)^2 \quad (2)$$

where $\hat{\mathbf{Y}}$ is the reconstruction of the RGB image $\mathbf{Y}$ of height h and width w. Models pretrained with this objective are denoted as $\langle +P_{\text{AE}}, \dots \rangle$.

The first classification objective, which we dub "*category-wise*", corresponds to predicting which of the 2^5 categories the input image corresponds to,[3] and is learned using a cross-entropy loss:

$$\mathcal{L}_{\text{CW}} = -\sum_{i=1}^{2^5} \mathbb{1}_{\{i=y\}} \log \hat{\mathbf{y}}_i \quad (3)$$

where $\hat{\mathbf{y}}$ is the vector $\big(p(y=1|I), \dots, p(y=2^5|I)\big)$ corresponding to the classifier's probability distribution over possible labels. Models pretrained with this objective are denoted as $\langle +P_{\text{CW}}, \dots \rangle$.

The second classification objective, called "*feature-wise*", consists in predicting each of the 5 feature values of the input image—i.e., an agreggate of five binary classification sub-tasks. The loss function for this last objective $\mathcal{L}_{\text{FW}}$ is thus:

$$\mathcal{L}_{\text{FW}} = -\sum_{f=1}^{5} \sum_{i=1}^{2} \mathbb{1}_{\{i=\mathbf{y}_f\}} \log \hat{\mathbf{Y}}_{fi} \quad (4)$$

where $\hat{\mathbf{Y}}$ is the structured prediction, such that $\hat{\mathbf{Y}}_{fi}$ is the probability assigned for the i^{th} possible value of the f^{th} feature, and $\mathbf{y} = (y_1, \dots, y_f)$ is the vector of target feature values for this example. We denote models pretrained with this objective as $\langle +P_{\text{FW}}, \dots \rangle$.

We also consider whether or not to *freeze* the parameters of the image encoder convolution stacks. Assuming the pretraining was successful, the resulting image vector representations should contain all the information necessary for models to succeed. In this case, freezing convolutions reduces the number of learnable parameters, which may help the optimization. Pretrained models whose convolution stacks are frozen are denoted as $\langle +P, +F, \dots \rangle$, whereas models whose convolutions (pretrained or not) are updated are denoted as $\langle \dots, -F, \dots \rangle$.

[3] Because the training split is used during pretraining, only the 2^4 base categories are in fact seen at this stage.

Distractor sampling. By default, during training, we first select the original/target category c_t uniformly at random, before selecting the distractor category c_d uniformly among remaining categories. A second strategy that we envision to improve performance consists in *adversarially sampling* c_d instead. More precisely, when we evaluate the agents at the end of each training epoch, we derive count-based estimates of the probability $P\left(\text{fail} \mid (c_t, c_d)\right)$ of communication failure for each pairs (c_t, c_d). At training time, c_d is sampled with a probability proportional to $P\left(\text{fail} \mid (c_t, c_d)\right)$. At evaluation time, c_d is still sampled uniformly. We denote the use of this adversarial sampling during training as $\langle \dots, +A, \dots \rangle$, and its absence as $\langle \dots, -A, \dots \rangle$.

Rewards and regularization. One drawback of the pretraining methods and the adversarial sampling alike is that most of them (i.e., all except the auto-encoder method) require information which might not be available in other datasets, namely labels pertaining to the semantics of the images.

One possible technique not subject to this concern consists in adding an *entropy* term to the REINFORCE loss, as is sometimes done in emergent communication (e.g., Lazaridou et al., 2018; Chaabouni et al., 2019a). This entropy loss is defined as:

$$\mathcal{L}_H = -\beta_S \sum_t H_{S,t} - \beta_R H_R \quad (5)$$

where β_S and β_R are two scalar coefficients controlling the strength of this regularization, $H_{S,t}$ is the entropy of the probability distribution computed by the sender and used to select the t^{th} symbol of the message, and H_R is the entropy of the probability distribution computed by the receiver. The scalar coefficients are set to $\beta_S = 10^{-2}$ and $\beta_R = 10^{-3}$.[4] The use of this entropy term is denoted with $\langle \dots, +H, \dots \rangle$.

Another technique consists in redefining the rewards system. Instead of associating each action of an episode with a binary reward $r \in \{0, 1\}$, the reward is defined as the probability that the receiver assigns to the target image, i.e., how confident it is in retrieving the target. The use of this confidence-based reward system is denoted with $\langle \dots, +C, \dots \rangle$.

The last technique that we study consists in deducting the recent average rewards as a *baseline*

[4] Optimal settings in preliminary experiments.

term b (Sutton and Barto, 2018, §13):

$$\mathcal{L} = -(r - b) \sum_t \log p(a_t) \tag{6}$$

where b is the average of r over the last 1000 batches. The use of this baseline term is denoted with $\langle \dots, +B \rangle$.

While confidence-based rewards and baseline can technically be applied jointly, doing so proves to be detrimental. None of the runs for models implemented as $\langle \dots, +C, +B \rangle$ yielded a successful communication protocol. We conjecture that this is due to the probability mass assigned to the target image being very close to the average reward (0.5) at the beginning of the training process, which leads to losses and gradient updates close to 0. In what follows, the use of these two techniques are then considered mutually exclusive.

Comparison with previous work. In our experiments, we exhaustively evaluate various design choices, which cover many architectures similar to those studied in earlier works. For instance, Lazaridou et al. (2018) would correspond to a $\langle -P, -F, -A, -H, -E, -B \rangle$ model, Bouchacourt and Baroni (2018) adopt a model similar to a $\langle +P_{cw}, +F, -A, -H - E, -B \rangle$. In what follows, we do not focus on how specific earlier works fare, but instead attempt to develop a more global picture.

5 Automatic metrics

Communication efficiency. We primarily measure the performance of a model by its *communication efficiency* (c.e.), which we define as the average probability assigned by the model to the target image over a large number of evaluation instances.[5]

Evaluation instances involve all categories seen during training with additional categories as well (see Section 3). To assess how the agents handle unseen combination of features at a finer level, we define *base-c.e.*, *gen.-c.e.* and *mixed-c.e.* by restricting the two selected categories to two base categories, two generalization categories, and one of each respectively.

All of our metrics are generalized from single models to sets of models by computing their average across models (i) using, for each model, the value obtained during the evaluation phase in which it reaches its highest communication efficiency and (ii) discarding any model which never reaches a communication efficiency of 60% or above at any point of the training process.[6] Any model that does reach a communication efficiency of 60% or above is said to be "successful". The *convergence ratio* (cvg.) of a set of models is the proportion of successful models in this set.

Abstractness. We task receivers with recognizing not the original image I shown to senders, but another target I' of the same category. This is meant to encourage senders to describe not so much the input image as its category. We evaluate this aspect using the *abstractness* of a model:

$$\text{abstractness} = 2 \cdot p_R(I'|I, I', m) \tag{7}$$

where $p_R(J)$ is the probability assigned by the receiver to the image J, I and I' are the original and target images, and m is the sender's message for the input I. Abstractness is 0 if all the mass is on the original image, and 1 when it is distributed evenly.[7]

Scrambling resistance. To measure how sensitive to symbol ordering receivers are, we define the *scrambling resistance* of a model by comparing the probability assigned to the target image by the receiver when provided with the sender's message m, and when provided with a randomly permuted version m' of it. More precisely, given a message m, we compute:

$$\begin{aligned} m &= (a_1, \dots, a_n) \\ m' &= \left(a_{\sigma(1)}, \dots, a_{\sigma(n)}\right) \\ \text{sr} &= \frac{\min\left(p_R(m),\ p_R(m')\right)}{p_R(m)} \end{aligned} \tag{8}$$

where a_t is the t^{th} symbol of the message produced by the sender, $p_R(x)$ is the probability of the receiver selecting the target image given the message

[5] Communication efficiency differs from *accuracy*, defined as the proportion of evaluation instances for which the target image is assigned a higher probability than the distractor. Accuracy can be maximal (100%) even with a very low communication efficiency ($50 + \epsilon$%). Low communication efficiency is a sign of sub-optimal performance, as an effective communication system should describe the target category unambiguously, i.e., the agents should solve the game with a high degree of confidence. In practice, we find these two values to be highly correlated in our experiments, suggesting our models are well calibrated (Guo et al., 2017).

[6] Such models are discarded because we are interested in the properties of emergent languages, i.e., communication protocols that are reliably used to convey information.

[7] As expected, we do not observe any value significantly larger than 1.

Implementation	cvg.	c.e.
$\langle -P, \ldots, -C, -B \rangle$	0.800	0.950
$\langle +P, -F, \ldots, -C, -B \rangle$	0.883	0.954
$\langle +P, +F, \ldots, -C, -B \rangle$	1.000	0.922
$\langle -P, \ldots, +C, -B \rangle$	0.875	0.954
$\langle +P, -F, \ldots, +C, -B \rangle$	0.958	0.961
$\langle +P, +F, \ldots, +C, -B \rangle$	1.000	0.926
$\langle -P, \ldots, -C, +B \rangle$	0.925	0.967
$\langle +P, -F, \ldots, -C, +B \rangle$	1.000	0.971
$\langle +P, +F, \ldots, -C, +B \rangle$	1.000	0.936

Table 1: Effects of pretraining and reward redefinition on convergence and communication efficiency.

x, and σ is a random permutation of the interval $[\![1, n]\!]$. The scrambling resistance of a model is an average of sr over a large number of evaluation instances.

Semantic probes. In order to determine which features of the original/target category are described in a sender's message, we implement a probing method based on decision trees. We convert any message m into a bag-of-symbols vector $u \in \mathbb{N}^{16}$, such that u_i is the number of occurrences of symbol i in m. Given a set of messages each associated with its corresponding original/target category, for each of the five features, we can train a decision tree to predict the values of the feature based on the bag-of-symbols representation of the messages. While the messages may very well encode information under a form that cannot be decoded by such a simple system, high accuracy from a decision tree is proof that the corresponding feature is consistently described in the messages.[8]

6 Results

6.1 Global performance

Table 1 shows the performance of all of the runs we have performed, aggregated based on the reward system they use (binary rewards, **confidence-based reward**, or binary rewards with a **baseline term**), on whether the visual convolution stacks are **pretrained** (without differentiating between the various pretraining objectives) and, if so, on whether these convolution stacks are **frozen** during training. We observe that the most impactful implementation choice is whether or not to use a baseline term ($\langle \ldots, -C, +B \rangle$). Improvements with $+B$ are much more consistent and pronounced than models using confidence-based rewards ($\langle \ldots, +C, -B \rangle$) or pretraining ($\langle +P, \ldots \rangle$).

[8] In this text we focus on the accuracy of the decision trees and leave the analysis of the trees themselves to future work.

Implementation	cvg.	c.e.
$\langle \ldots, -H, \ldots \rangle$	0.929	0.941
$\langle \ldots, +H, \ldots \rangle$	0.988	0.952
$\langle \ldots, {}_{-F}, \ldots, -H, {}_{-C}, {}_{+B} \rangle$	1.000	0.970
$\langle \ldots, {}_{-F}, \ldots, +H, {}_{-C}, {}_{+B} \rangle$	0.963	0.970

Table 2: Effects of the entropy loss term on convergence and communication efficiency.

On its own, pretraining brings some degree of improvement comparable to what we see in models implemented as $\langle \ldots, +C, -B \rangle$. Setups involving freezing pretrained convolution stacks ($\langle +P, +F, \ldots \rangle$) reach a convergence ratio of 1 at the expense of a downgrade in communication efficiency. Moreover, pretraining without freezing weights ($\langle +P, -F, \ldots \rangle$), while not detrimental, does not improve performances unless used jointly with either $+C$ or $+B$. Optimal performances are attested when using pretraining with a baseline term ($\langle +P, -F, \ldots, -C, +B \rangle$).

Table 2 shows the performance (top) of all of the runs that we have performed and (bottom) of all runs with the baseline term and without frozen convolution stacks, aggregated based on whether they are trained with the **entropy penalty**. We observe that, while in general using this regularization term is an efficient way to boost both the convergence ratio and the communication efficiency of converging runs, this positive effect does not persist with $\langle \ldots, -F, \ldots, -C, +B \rangle$ runs (see below for more information about the drop in cvg. in this case).

Because of their high performance, we focus on models implemented as $\langle \ldots, -F, \ldots, -C, +B \rangle$ in the remainder of this discussion. A communication efficiency around 97% might intuitively seem an indicator of excellent performance, but remark that, should the sender completely ignore one semantic feature of the images, then the communication efficiency could still rise up to $\frac{30.5}{31}$ ($\approx 98.4\%$): this value is obtained when, among the 31 possible categories for the distractor, 30 lead to perfect retrieval of the target image and 1 leads to chance retrieval. As such, none of the performances seen so far guarantees that all features are encoded in the messages.

Implementation	cvg.	c.e.
$\langle -P, -F, -A, \ldots, -C, +B \rangle$	1.000	0.958
$\langle -P, -F, +A, \ldots, -C, +B \rangle$	0.850	0.978
$\langle +P, -F, -A, \ldots, -C, +B \rangle$	1.000	0.959
$\langle +P, -F, +A, \ldots, -C, +B \rangle$	1.000	0.983
$\langle -P, -F, +A, -H, -C, +B \rangle$	1.000	0.981
$\langle -P, -F, +A, +H, -C, +B \rangle$	0.700	0.974

Table 3: Effects of adversarial sampling. The two last lines are a decomposition of the second one.

Implementation	cvg.	c.e.
$\langle -P, -F, +A, \ldots, -C, +B \rangle$	0.859	0.978
$\langle +P_{\text{AE}}, -F, +A, \ldots, -C, +B \rangle$	1.000	0.981
$\langle +P_{\text{CW}}, -F, +A, \ldots, -C, +B \rangle$	1.000	0.985
$\langle +P_{\text{FW}}, -F, +A, \ldots, -C, +B \rangle$	1.000	0.983

Table 4: Effects of pretraining objectives

Table 3 shows the performance of the runs aggregated based on the **sampling strategy** for distractors and the use of **pretraining** for the visual convolution stacks (still without differentiating between the various pretraining objectives). We see that, compared to uniform sampling, the adversarial sampling strategy systematically and substantially increases the communication efficiency. Nonetheless, the adversarial strategy can induce a lower convergence ratio when the convolution stacks are not pretrained and an entropy penalty is added, suggesting that this sampling strategy and the entropy penalty used jointly make training too challenging for agents with randomly initialized convolution stacks. In all, the higher performances observed with the adversarial sampling strategy lead us to narrow down our discussion once more, this time focusing on models implemented as $\langle \ldots, -F, +A, \ldots, -C, +B \rangle$.

Finally, we focus on the effect of the different **pretraining** objectives in Table 4. Though all three pretraining objectives are helpful, we observe the highest improvement in communication efficiency with the two classification objectives. Among them, the category-wise objective outperforms the feature-wise objective. While the feature-wise objective provides feature-level guidance, the category-wise pretraining regimen directly trains the convolution stacks to tease apart images of different categories, which is what the signaling game requires of them. We hypothesize that the feature-wise objective might be superior when the category space is sufficiently larger and more complex.

6.2 Generalization and language analysis

Having looked at how to foster reliability and high performance, we now turn to how to a study of how well the models generalize to unseen items and whether their messages display language-like characteristics—as the literature often remarks that such characteristics should not be taken for granted (Mu and Goodman, 2021; Patel et al., 2021).

Abstractness. Abstractness is systematically close to 1. Over all 805 successful runs, it averages to 0.992 ($\sigma = \pm 0.015$). On the 77 successful $\langle \ldots, -F, +A, \ldots, -C, -B \rangle$ runs, it reaches 0.996 ($\sigma = \pm 0.008$), with no statistically significant difference between the four pretraining options. In all, using distinct images as original and target inputs does induce the senders to describe categories rather than specific images.

However, when grouping runs implemented as $\langle \ldots, +A, \ldots, -C, -B \rangle$ depending on their pretraining and convolution freezing, we find one group of outliers: $\langle P_{\text{AE}}, +F, +A, \ldots, -C, -B \rangle$ runs have an abstractness of 0.958. This value is statistically lower than for each of the six other groups (as shown by a Pitman test; $p < 10^{-6}$ in all cases). Convolution stacks pretrained as auto-encoders learn to capture the specificity of each image, which apparently permeates the emergent languages if subsequently frozen.

We also observe an opposite—albeit weaker—effect with the category-wise pretraining objective. $\langle P_{\text{CW}}, +F, +A, \ldots, -C, -B \rangle$ runs have an abstractness of 0.998, higher than the 0.994 of $\langle P_{\text{CW}}, -F, +A, \ldots, -C, -B \rangle$ runs. The difference ($p < 0.04$, Pitman test) indicates that in such cases, fine-tuning the convolution stacks leads the agents to include image-specific information in their messages.

Scrambling resistance. Scrambling resistance yields high values, ranging from 0.892 when using auto-encoder pretraining to 0.915 when using feature-wise pretraining.[9] In other words, the receiver is able to recognize a category based on a randomly permuted message with a high degree of accuracy. This property, however, does not entail that the sender produces symbols in a (near) random order. Indeed, even English, which requires a

[9] The difference between these two pretraining regimens is statistically significant: $p < 10^{-3}$ (Pitman permutation test).

rather strict word-order, arguably has a high scrambling resistance: it is natural to associate the scrambled sentence "cube a there blue is" with a picture of a blue cube rather than that of a blue sphere (or a red cube, etc.). High scrambling resistance points towards the possibility that each symbol is loaded with an intrinsic meaning, the interpretation of which is fairly independent of its position—in contrast with, e.g., the digits in positional numeral systems (which are compositional systems with low scrambling resistance).

Generalization. As we saw in Section 6.1, the highest communication efficiency we observe, of 0.985, is obtained with the $\langle P_{\text{CW}}, -F, +A, \ldots, -C, -B\rangle$ implementation. Let us recall that this means that when the source/target category and the distractor category are selected from the whole set of categories, the receiver puts on average 0.985 of the probability mass of its choice distribution on the target image. As for the base-c.e. (when both categories are base categories, i.e., not seen during training) of this implementation, its value is near perfect, above 0.999. Its gen-c.e. (when both categories are generalization categories), is also very high, at 0.997. These different values indicate that the models are able to generalize very well not only to unseen images but also to new categories (i.e., unseen combinations of features).

For this same implementation, the mixed-c.e. (when only one of the categories is a base category) drops to 0.971.[10] Recall that this is the only case where target and distractor may differ by a single feature. Even if agents disregard one feature, their mixed-c.e. can still theoretically reach up to $\frac{14.5}{15}$ ($\approx 96.7\%$). Hence, $\langle P_{\text{FW}}, -F, +A, \ldots, -C, -B\rangle$ runs communicate about all features, despite it not being required by the training objective. Similarly, $\langle P_{\text{CW}}, -F, +A, \ldots, -C, -B\rangle$ runs obtain a mixed-c.e. of 0.967 (almost equal to the threshold) and $\langle -P, -F, +A, \ldots, -C, -B\rangle$ runs reach a mixed-c.e. of 0.964 (slightly below).

Semantic content Scrambling resistance scores highlight that the semantic contents of symbols are mostly position-insensitive. This entails that our decision-tree based probes, which rely on bag-of-symbols representations of the messages, are relevant. Table 5 shows how shape is much less accurately conveyed than other image features.[11] This indicates that shape is harder to identify than color, size or position and that since the training process does not incentivize the agents to describe all features, they systematically focus on the four easiest.[12]

Implementation	color	shape
$\langle\ -P,\ {}_{-F},\ {}_{+A},\ -H,\ {}_{-C},\ {}_{-B}\ \rangle$	0.992	0.534
$\langle +P_{\text{AE}},\ {}_{-F},\ {}_{+A},\ -H,\ {}_{-C},\ {}_{-B}\ \rangle$	0.962	0.558
$\langle +P_{\text{CW}},\ {}_{-F},\ {}_{+A},\ -H,\ {}_{-C},\ {}_{-B}\ \rangle$	0.999	0.532
$\langle +P_{\text{FW}},\ {}_{-F},\ {}_{+A},\ -H,\ {}_{-C},\ {}_{-B}\ \rangle$	0.993	0.537
$\langle\ -P,\ {}_{-F},\ {}_{+A},\ +H,\ {}_{-C},\ {}_{-B}\ \rangle$	0.972	0.595
$\langle +P_{\text{AE}},\ {}_{-F},\ {}_{+A},\ +H,\ {}_{-C},\ {}_{-B}\ \rangle$	0.988	0.656
$\langle +P_{\text{CW}},\ {}_{-F},\ {}_{+A},\ +H,\ {}_{-C},\ {}_{-B}\ \rangle$	1.000	0.617
$\langle +P_{\text{FW}},\ {}_{-F},\ {}_{+A},\ +H,\ {}_{-C},\ {}_{-B}\ \rangle$	0.999	0.598

Table 5: Decision tree classifiers: feature prediction accuracy (color and shape).

Interestingly, applying an entropy penalty during training strongly drives the agents to communicate about the shape. Moreover, models pretrained with the auto-encoder objective lead to higher values than any others.[13] The difference in shape recognition between this group and the others is always significant ($p < 10^{-2}$).

7 Conclusions

Two broad conclusions emerge from our experiments. Firstly, we saw that not all implementations perform equally well. We demonstrated how the use of a baseline term or an adversarial input sampling mechanism were necessary to reach high performance. While pretraining convolution stacks can prove beneficial in limited circumstances, not fine-tuning them afterwards may prove to be highly detrimental. In all, a well designed implementation can learn reliably and generalize to new images and combinations of features.

Secondly, we have made a case for the need of fine-grained methods when analyzing the emergent communication protocol. We have introduced an

[10] In this case, receivers marginally favor the image from a base category.

[11] The three remaining features being very much in line with color, we omit them in this table for brevity and clarity. See the full results in Table 7 in Appendix B.2.

[12] An unlikely alternative is that they communicate the shape of the object in a complex way that is mostly inaccessible to our decision trees.

[13] The values are not shown here, but freezing pretrained convolution stacks does not improve (and in fact deteriorates) the accuracy of the shape-probing decision trees, except for the auto-encoder objective.

array of tools. Among them, scrambling resistance were used to demonstrate that each symbol in our languages has semantic contribution independent from its position. Decision trees based probes informed us that these symbols were put to use to systematically describe all but one of the input image's features, shape being constitently neglected though not entirely ignored despite the possibility we left open through the design of the training instances. These results also connect with design choices: for instance, we saw how entropy regularization and auto-encoder pretraining strengthened the prominence of shape in the messages.

We next plan to experiment with a partition of categories between base and generalization that forces all features to be encoded in the messages, and then use decision trees and other methods to automatically describe the syntax and the semantics of the emergent communication protocols in simple terms, so as to better characterize how these protocols relate to natural language. We also plan to study the impacts of the semantic complexity of the input images on these emergent protocol, using a richer set of features and values, and using unlabeled real-world scenes. Lastly, our findings will have to be confirmed in setups involving other games such as navigation tasks.

Limitations

There are two main limitations to the present work. First and foremost is the computational cost associated with the present experiments. We present here results and analyzed gleaned over 10 runs, 7 pretraining regimens, 8 RL gradient propagation variants and 2 data sampling approaches, for a total of 1120 models. While training any one of our models is cheap (less than 3 hours on a single A100 NVIDIA GPU), the total number of models may pose a challenge for future replication studies and comes at an environmental cost. This also prevented us from selecting optimal batch size, learning rate, and so on for specific setups—as described in Appendix A, we set these values globally prior to running experiments. This may affect results and impact conclusions.

Second is the theoretical scope of the current paper. We have focused solely on single-turn, 2 agents signaling game setups. The recommendations and conclusions drawn in the present paper may or may not translate to other language games. Likewise, while this study aims at exhaustiveness, material limitations have bounded the scope of implementation choices we studied. Some approaches, such as KL regularization (Geist et al., 2019), have thus been left out of the present study.

Acknowledgments

The authors deeply thank Takamura Hiroya who participated in preliminary experiments related to this work.

Preliminary results were obtained from project JPNP15009, commissioned by the New Energy and Industrial Technology Development Organization (NEDO), using the computational resources of the AI Bridging Cloud Infrastructure (ABCI), provided by the National Institute of Advanced Industrial Science and Technology (AIST), Japan.

This work is part of the FoTran project, funded by the European Research Council (ERC) under the EU's Horizon 2020 research and innovation program (agreement № 771113). We also thank the CSC-IT Center for Science Ltd., for computational resources.

This work was also supported by an Émergence 2021 grant (SYSNEULING project) from IdEx Université Paris Cité, as well as a public grant overseen by the French National Research Agency (ANR) as part of the "Investissements d'Avenir" program: IdEx *Lorraine Université d'Excellence* (reference: ANR-15-IDEX-0004).

References

Jacob Andreas. 2019. Measuring compositionality in representation learning. In *International Conference on Learning Representations*.

Marco Baroni. 2019. Linguistic generalization and compositionality in modern artificial neural networks. *Philosophical Transactions of the Royal Society B*, 375.

Diane Bouchacourt and Marco Baroni. 2018. How agents see things: On visual representations in an emergent language game. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 981–985, Brussels, Belgium. Association for Computational Linguistics.

Henry Brighton and Simon Kirby. 2006. Understanding linguistic evolution by visualizing the emergence of topographic mappings. *Artif. Life*, 12(2):229–242.

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*, pages 4427–4442, Online. Association for Computational Linguistics.

Rahma Chaabouni, Eugene Kharitonov, Emmanuel Dupoux, and Marco Baroni. 2019a. Anti-efficient encoding in emergent communication. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Rahma Chaabouni, Eugene Kharitonov, Alessandro Lazaric, Emmanuel Dupoux, and Marco Baroni. 2019b. Word-order biases in deep-agent emergent communication. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5166–5175, Florence, Italy. Association for Computational Linguistics.

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*.

Katrina Evtimova, Andrew Drozdov, Douwe Kiela, and Kyunghyun Cho. 2018. Emergent communication in a multi-modal, multi-step referential game. In *International Conference on Learning Representations*.

Jakob Foerster, Ioannis Alexandros Assael, Nando de Freitas, and Shimon Whiteson. 2016. Learning to communicate with deep multi-agent reinforcement learning. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 2137–2145. Curran Associates, Inc.

Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. 2019. A Theory of Regularized Markov Decision Processes. In *ICML 2019 - Thirty-sixth International Conference on Machine Learning*, Long Island, United States. ICML 2019.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On Calibration of Modern Neural Networks. In *International Conference on Machine Learning*, pages 1321–1330. PMLR. ISSN: 2640-3498.

Serhii Havrylov and Ivan Titov. 2017. Emergence of language with multi-agent games: Learning to communicate with sequences of symbols. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. 2012. Coursera lectures slides, lecture 6.

Charles F. Hockett. 1960. The origin of speech. *Scientific American*, 203(3):88–96.

Emilio Jorge, Mikael Kågebäck, and Emil Gustavsson. 2016. Learning to play guess who? and inventing a grounded language as a consequence.

Jooyeon Kim and Alice Oh. 2021. Emergent communication under varying sizes and connectivities. In *Advances in Neural Information Processing Systems*, volume 34, pages 17579–17591. Curran Associates, Inc.

Simon Kirby. 2002. Natural language from artificial life. *Artif Life*, 8(2):185–215.

Simon Kirby, Hannah Cornish, and Kenny Smith. 2008. Cumulative cultural evolution in the laboratory: An experimental approach to the origins of structure in human language. *Proceedings of the National Academy of Sciences*, 105(31):10681–10686.

Simon Kirby, Tom Griffiths, and Kenny Smith. 2014. Iterated learning and the evolution of language. *Curr. Opin. Neurobiol.*, 28:108–114.

Tomasz Korbak, Julian Zubek, Lukasz Kucinski, Piotr Milos, and Joanna Raczaszek-Leonardi. 2019. Developmentally motivated emergence of compositional communication via template transfer. *CoRR*, abs/1910.06079.

Angeliki Lazaridou, Karl Moritz Hermann, Karl Tuyls, and Stephen Clark. 2018. Emergence of linguistic communication from referential games with symbolic and pixel input. In *International Conference on Learning Representations*.

Angeliki Lazaridou, Alexander Peysakhovich, and Marco Baroni. 2017. Multi-agent cooperation and the emergence of (natural) language. In *International Conference on Learning Representations*.

David Lewis. 1969. *Convention: a philosophical study*. Harvard University Press Cambridge.

Paul Pu Liang, Jeffrey Chen, Ruslan Salakhutdinov, Louis-Philippe Morency, and Satwik Kottur. 2020. On emergent communication in competitive multi-agent teams. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '20, page 735–743, Richland, SC. International Foundation for Autonomous Agents and Multiagent Systems.

Timothee Mickus, Timothée Bernard, and Denis Paperno. 2020. What meaning-form correlation has to compose with: A study of MFC on artificial and natural language. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3737–3749, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Daniela Mihai and Jonathon Hare. 2021. Learning to draw: Emergent communication through sketching. In *Advances in Neural Information Processing Systems*, volume 34, pages 7153–7166. Curran Associates, Inc.

Jesse Mu and Noah Goodman. 2021. Emergent communication of generalizations. In *Advances in Neural Information Processing Systems*, volume 34, pages 17994–18007. Curran Associates, Inc.

M. A. Nowak, J. B. Plotkin, and D. Krakauer. 1999. The evolutionary language game. *Journal of Theoretical Biology*, 200(2):147–162.

Shivansh Patel, Saim Wani, Unnat Jain, Alexander G. Schwing, Svetlana Lazebnik, Manolis Savva, and Angel X. Chang. 2021. Interpretation of emergent communication in heterogeneous collaborative embodied agents. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15953–15963.

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*.

Mathieu Rita, Rahma Chaabouni, and Emmanuel Dupoux. 2020. "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. Association for Computational Linguistics.

Mathieu Rita, Corentin Tallec, Paul Michel, Jean-Bastien Grill, Olivier Pietquin, Emmanuel Dupoux, and Florian Strub. 2022. Emergent communication: Generalization and overfitting in lewis games. In *Advances in Neural Information Processing Systems*.

Sainbayar Sukhbaatar, Arthur Szlam, and Rob Fergus. 2016. Learning multiagent communication with backpropagation. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.

Richard S. Sutton and Andrew G. Barto. 2018. *Reinforcement Learning: An Introduction*, second edition edition. Adaptive Computation and Machine Learning series. MIT Press.

Ronald J. Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.*, 8(3–4):229–256.

Kaiqing Zhang, Zhuoran Yang, and Tamer Başar. 2021. *Multi-Agent Reinforcement Learning: A Selective Overview of Theories and Algorithms*, pages 321–384. Springer International Publishing, Cham.

A Hyperparameters selection and training details

Throughout our experiments, we allow agents to generate messages of up to 10 symbols long, using a vocabulary of 16 symbols. We train all models for up to 100 epochs of 1000 batches each, using 128 training instance per batch. We repeat each training procedure across 10 random seeds. Parameters are optimized with RMSProp (Hinton et al., 2012).

Prior to any experiment reported here, we ran a small-scale grid-search to select a learning rate most likely to reliably induce a successful emergent communication protocol. We exhaustively test learning rates in $\{10^{-x/2} \mid 4 \leq x \leq 12\}$and measure the convergence ratio for groups of 10 runs trained for 50 epochs. Results, displayed in Figure 2, suggest an optimal learning rate of 10^{-4} which we adopt in all subsequent experiments.[14]

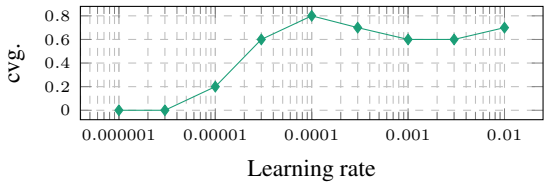

Figure 2: Convergence ratio as a function of learning rate.

In Section 4, hyperparameter values for the pretraining procedures were selected based on the models' lack of further improvement on a held-out subset of the training data. Using 1000 steps per epoch and batches of 128 images, we found that 5 epochs and a learning rate of $3 \cdot 10^{-4}$ was sufficient to guarantee an accuracy close to 100% for the classification pretraining tasks, whereas the auto-encoding task required 40 epochs with the same learning rate.

B Supplementary results

B.1 Meaning–Form Correlation

In compositional languages, the meaning and the form of messages tend to be correlated: Minute changes in form (e.g., substitutions of a single token) are expected to correspond to minute changes in meaning. To study the compositionality of the communication protocols set up by the agents, one can also measure their *meaning-form correlation* (MFC).

Meaning-form correlation, or topological similarity, consists in comparing how the distance between two messages relates to the distance between their semantic contents. More formally, it is computed as a Spearman correlation between two paired samples of distance measurements $D_F = (d_F(o_i, o_j))_{1 \leq i < j \leq n}$ and $D_M = (d_M(o_i, o_j))_{1 \leq i < j \leq n}$ over the same set of observations, with the assumption that one distance function (d_F) captures variation in form and the other (d_M) capture variation in meaning. For clarity, we

[14] Learning rates greater than 0.0003 yield to unstable performances, with some runs reverting back to chance-level communication efficiency.

denote an MFC correlation score using the symbol τ. In our case, we have compared the Jaccard index of the two messages as bags-of-symbols to the Hamming distance between the two corresponding image categories. [15]

MFC scores are not easy to interpret by themselves, but it can be illuminating to see how they vary and correlate with properties. While the distribution of MFC and its relation with communication efficiency is quite complex, we have observed that difficult setups (e.g., where a globally useful design choice is not implemented, or where an adversarial sampling strategy factors in) display two trends: on the one hand, they exhibit lower MFC scores, on the other hand, for such a setup, the MFC scores of individual runs are more in line with with performance (i.e., they display a stronger Spearman correlation or a weaker anti-correlation with communication efficiency).

Implementation	MFC	corr. with c.e.	
	τ	ρ	p
$\langle \ldots, {}_{-F}, \quad \ldots, \quad -B \rangle$	0.348	-0.157	< 0.008
$\langle \ldots, {}_{-F}, \quad \ldots, \quad +B \rangle$	0.388	-0.237	< 0.003
$\langle \quad \ldots, \quad +A, \quad \ldots \quad \rangle$	0.328	0.396	$< 3 \cdot 10^{-16}$
$\langle \quad \ldots, \quad -A, \quad \ldots \quad \rangle$	0.351	0.262	$< 9 \cdot 10^{-8}$
$\langle \ldots, {}_{-F}, +A, \quad \ldots, \quad {}_{+B} \rangle$	0.375	0.168	0.143
$\langle \ldots, {}_{-F}, -A, \quad \ldots, \quad {}_{+B} \rangle$	0.400	-0.385	< 0.0005

Table 6: Some MFC scores and correlations with communication efficiency.

For example, the two top rows of Table 6 show a case in which the absence of a baseline term entails a lower MFC and a weaker anti-correlation with c.e. The middle two rows show a case in which the use of the adversarial distractor sampling strategy during training also entails a lower MFC and a stronger correlation with c.e. The two bottom rows show another case in which the adversarial training strategy has a similar effect. In addition, the last row shows that when the training is made particularly easy, the models produce on average messages that are very compositional (in the sense reflected by the MFC), but that the best models diverge from this: the best models are the ones in which the two agents develop some form of co-adaptation at odds with compositionality. This echoes the findings of Chaabouni et al. (2020), who highlight that MFC is not necessarily tied to generalization capabilities.

[15] Using the Levenshtein distance instead of the Jaccard index yields the same conclusions, as MFC scores derived from either distance are extremely significantly correlated.

B.2 Decision Trees

Full results for the decision-tree semantic content probes are displayed in Table 7. As noted in the main text, the behavior for size and position features is very similar to that for color, and very distinct from that for shape.

Implementation	color	size	h-pos	v-pos	shape
$\langle\ -P,\ {}_{-F},\ {}_{+A},\ -H,\ {}_{-C},\ {}_{-B}\ \rangle$	0.992	0.964	0.992	0.998	0.534
$\langle +P_{\text{AE}},\ {}_{-F},\ {}_{+A},\ -H,\ {}_{-C},\ {}_{-B}\ \rangle$	0.962	0.974	0.979	0.986	0.558
$\langle +P_{\text{CW}},\ {}_{-F},\ {}_{+A},\ -H,\ {}_{-C},\ {}_{-B}\ \rangle$	0.999	0.998	0.987	0.987	0.532
$\langle +P_{\text{FW}},\ {}_{-F},\ {}_{+A},\ -H,\ {}_{-C},\ {}_{-B}\ \rangle$	0.993	0.968	0.993	0.993	0.537
$\langle\ -P,\ {}_{-F},\ {}_{+A},\ +H,\ {}_{-C},\ {}_{-B}\ \rangle$	0.972	0.958	0.968	0.968	0.595
$\langle +P_{\text{AE}},\ {}_{-F},\ {}_{+A},\ +H,\ {}_{-C},\ {}_{-B}\ \rangle$	0.988	0.992	0.991	0.984	0.656
$\langle +P_{\text{CW}},\ {}_{-F},\ {}_{+A},\ +H,\ {}_{-C},\ {}_{-B}\ \rangle$	1.000	0.999	1.000	1.000	0.617
$\langle +P_{\text{FW}},\ {}_{-F},\ {}_{+A},\ +H,\ {}_{-C},\ {}_{-B}\ \rangle$	0.999	0.998	0.999	1.000	0.598

Table 7: Decision tree classifiers: feature prediction accuracy (all features).

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Section Limitations (after Section 7 Conclusions).

☐ A2. Did you discuss any potential risks of your work?
Not applicable. Left blank.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract and Section 1 Introduction.

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Section 3 Experimental setup, paragraph Dataset.

☑ B1. Did you cite the creators of artifacts you used?
Section 3 Experimental setup, paragraph Dataset.

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Not applicable. (I think it is not applicable as we haven't yet released the dataset that we are using.)

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Section 3 Experimental setup, paragraph Dataset.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Not applicable. Left blank.

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Section 3 Experimental setup, paragraph Dataset.

C ☑ Did you run computational experiments?

Section 3 Experimental setup and Section 4 Implementation choices.

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section Limitations (after Section 7 Conclusions).

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 3 Experimental setup and Appendix A Hyperparameters selection and training details.

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 6 Results.

☐ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Not applicable. Left blank.

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.