Noise May Drown Out Words but Foster Compositionality: The Advantage of the *Erasure* and *Deletion* Noisy Channels on Emergent Communication

Cezary Klamra[1], Francijn Keur[1] and Raquel G. Alhama[2]
[1] Master of Logic, Institute for Logic, Language and Computation, University of Amsterdam
[2] Institute for Logic, Language and Computation, University of Amsterdam
{cezary.klamra, francijn.keur}@student.uva.nl,rgalhama@uva.nl

Abstract

We investigate communication emerging in noisy environments with the goal of capturing the impact of message disruption on the emerged protocols. We implement two different noise mechanisms, inspired by the erasure and deletion channels studied in information theory, and simulate a referential game in a neural agent-based model with a variable message length channel. We leverage a stochastic evaluation setting to apply noise only *after* a message is sampled, which adds ecological validity and allows us to estimate information-theoretic measures of the emerged protocol directly from symbol probabilities. Contrary to our expectations, the emerged protocols do not become more redundant with the presence of noise; instead, we observe that certain levels of noise encourage the sender to produce more compositional messages, although the impact varies depending on the type of noise and input representation.

1 Introduction

Emergent Communication (EC) studies how artificial agents spontaneously develop their own language through interaction. Simulations with such systems offer insight into fundamental questions related to the origins of language and the biases (inductive or environmental) that are key for structure to emerge. EC frameworks also have a role in applied research, as they have been used for the automatic development of communication protocols in several real-world domains, such as autonomous driving, traffic control, or internet of things (Chafii et al., 2023; Pi et al., 2024).

The relevance of incorporating more naturalistic settings into EC has been increasingly recognized, as communication protocols emerging in overly artificial settings tend to lack crucial features of natural languages (Chaabouni et al., 2019a, 2020; Galke and Raviv, 2025). Simulations with agents endowed with either human-like cognitive constraints or biases (Kouwenhoven et al., 2024; Rita et al., 2022; Galke et al., 2022; Galke and Raviv, 2025; Chaabouni et al., 2019b) or a learning environment in which the agents are subject to naturalistic communication pressures (Lazaridou and Baroni, 2020; Galke et al., 2022) seem to encourage the presence of some of language-like properties, such as Zipfian distribution of message lengths, or the word-order/case-marking trade-off.

A possible way to make the setting more naturalistic is by incorporating a noisy environment. In the real world, message transmission is often imperfect, either because of competing sounds in the environment or due to the listener's perceptual or attentional constraints. To facilitate communication in noisy environments, a language must offer some degree of resilience to disruptions resulting from random distortions of the produced utterances. Redundancy and compositionality are two features of natural language that can help in that endeavor.

The former characterizes human communication on several levels (Aylett, 2000); for instance, redundancy of English was estimated to be around 50% at the letter level, according to the information-theoretic formalization of the notion (Shannon, 1951). Some phonetic features may also be considered redundant, particularly the distinction between an *allophone* and a *phoneme* (Bazzanella, 2011). Numerous ways of conveying the same meaning provide another example, e.g. consider the phrases: "*How old are you?*", "*What is your age?*", "*What age are you?*". Nonetheless, redundancy does not seem to grow indefinitely in (emergent) languages (Beekhuizen et al., 2013), as exemplified by the trade-off between word order and case markings (Chaabouni et al., 2019a; Lian et al., 2021).[1] On the

[1]This trade-off refers to the tendency of natural languages to rely either on case-marking with more flexible word-order (e.g. Russian), or stricter word-order with little or no case-marking (e.g. English) to encode the role of sentence con-

Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 3141–3166

other hand, compositionality divides the meaning of a message across multiple lexical pieces (Szabó, 2004). This has the potential to help preserve the correct decoding of a message: if the message "red hat" is altered due to noise but part of it is preserved ("red" or "hat") we may still infer the referent.

Here, we follow previous work (Kuciński et al., 2020, 2021) which use an Emergent Communication (EC) scenario to explore whether the presence of noise in a communication channel encourages a neural agent-based system to develop robust communication protocols via the use of redundancy and compositionality. Our contributions are as follows: (i) our agents are allowed to use variable-length messages; (ii) we study two types of noise –erasure and deletion– derived from information-theoretic models of noisy communication channels (Shannon, 1948; Mitzenmacher, 2009); (iii) we leverage a stochastic evaluation setting in which the sender samples symbols from the learned distribution, and we apply the noise only *after* the sender has sampled a message. This has multiple benefits: (1) the formalization of noise applies only to the channel and is therefore a more ecologically valid technique for simulating disruptions (compared to approaches which apply noise as a regularization layer before a message is sampled (Foerster et al., 2016; Kuciński et al., 2021), and (2) it allows us to compute a range of information-theoretic measures on the emergent protocols with greater reliability, compared to the deterministic approach.

We find that the presence of (certain amounts of) noise encourages compositionality but not redundancy of the messages generated by the sender. However, compositionality is not always preserved in the corrupted messages reaching the receiver.

2 Related Work

Simulating EC in a cooperative setting is often based on a referential game, most often a version of the Lewis Signaling Game (Lewis, 1969), in which two agents are trained to achieve a common goal: the Sender observes a state and generates a message, based on which the Receiver's goal is to select a unique correct state, i.e. the target, among a set of candidates; both are awarded if the correct state is chosen. Over repeated iterations of the game, the agents develop a shared communication protocol based on the learning signal.

In recent work on EC, the Sender and Receiver are modeled as neural network architectures (see Lazaridou and Baroni, 2020 and Peters et al., 2025 for an overview). Much focus has been put on studying the emergence of compositionality. While emerging communication protocols do not always exhibit this property (Lazaridou and Baroni, 2020; Peters et al., 2025), choices on model capacity and channel bandwidth (Gupta et al., 2020), input representations (Lazaridou et al., 2018; Słowik et al., 2020; Akkerman et al., 2024), model architecture and training regimes (Havrylov and Titov, 2017; Ren et al., 2020; Chaabouni et al., 2020; Galke and Raviv, 2025) seem to influence the emergence of compositional languages.

Another line of work investigates how noise influences agents' communication. Some of this work primarily focuses on how the presence of noise impacts task performance (Simões et al., 2019; Kontogiorgis and Bouroche, 2024; Weil et al., 2023). Other studies on EC in a noisy setting focus specifically on how it affects the protocols themselves, such as the presence of Zipf's law of abbreviation (Ueda and Washio, 2021), symbol distributions (Foerster et al., 2016), or properties facilitating zero-shot communication (Cope and Schoots, 2024). In the domain of information theory, Letizia et al. (2023) exploit a cooperative game setting inspired by GANs to learn capacity of noisy channels, i.e. the maximum possible rate at which information can be reliably transmitted.

Studies investigating compositionality or redundancy in noisy settings are particularly relevant to this work. Nikolaus (2024) studies explicit conversational mechanisms and observes that the presence of noise boosts compositionality, unless the setting includes a feedback mechanism facilitating conversational repair. Vital et al. (2025) find that communication emerging in noisy environments is more robust to disrupting messages by masking symbols, possibly due to more redundant protocols. Kuciński et al. (2021) apply noise to symbol distributions in Straight-Through Gumbel-Softmax based simulations. They find that moderate levels of noise promote compositionality and prove that under certain conditions, convergence to a compositional protocol is guaranteed in the presence of noise, noting that the effect may be specific to the fixed message-length setting they use.

stituents. However, languages rarely include both strategies, and so in that sense are not redundant.

3 Experimental Setting

Inspired by Lazaridou et al. (2018), we simulate the referential game with variable message length in two settings, utilizing either disentangled binary vectors or pixel data as the input dataset. We use the EGG framework (Kharitonov et al., 2019). Our code and other resources required to reproduce the reported results are made available at https://github.com/cklamra/noisyEC/.

3.1 Input Data

We use two complementary datasets: one using disentangled feature vectors, and another using pixel-based inputs:

Disentangled Input We follow the procedure described by Lazaridou et al. (2018) to recreate disentangled feature vectors from the Visual Attributes for Concepts Dataset (Silberer et al., 2013).[2] Each of the 503 sparse binary vectors represents 594 disentangled attributes of a single concept belonging to one of 16 categories. We randomly choose 402 vectors to be used for training and in-domain evaluation (ID) and use the remaining 101 vectors for out-of-domain evaluation (OOD). Then, for each target vector, we repeatedly sample sets 4 distractors.[3] The resulting dataset consists of 102912/1206/1212 samples for training, ID and OOD evaluation, respectively.

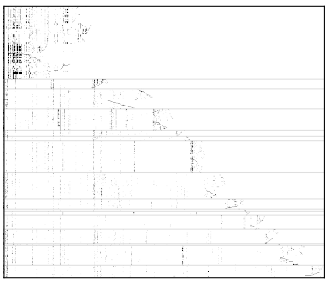


Figure 1: Sparsity pattern in the VisA dataset (Disentangled input). Each row represents a single concept, whose features are marked in black. Horizontal lines separate concept categories.

Pixel Input We generate 64×64 RGB images of single object scenes using a modified version of the Obverter dataset scripts (Choi et al., 2018). We consider 5 object shapes and 8 colors from the original dataset, and additionally include the *cone* shape. 12 out of the 48 resulting combinations are solely used for OOD evaluation, and the remaining 36 are used for training and ID evaluation. For each target scene, we sample 256/12/36 sets of 4 distractor scenes, resulting in 9216/432/432 total samples, for training, ID and OOD evaluation, correspondingly. Additional details are provided in Appendix A.

Although pixel input is generally considered more realistic and challenging (as agents need to learn the relevant attributes themselves), attribute distributions in the pixel input condition are in fact much more artificial: even though in some cases we constrain the set of possible candidates (see above), the features of the target object follow uniform distributions, whereas VisA attributes are sourced from the real world, thanks to which intra-concept features reflect subtle co-dependencies, as illustrated in Figure 1.

3.2 Architecture

Let A be the set of available symbols and $L_{\max}$ be the maximum message length. Given the target object $t = c_i$, where c_i belongs to the sequence of candidate objects $\mathcal{C} = (c_1, \ldots, c_5)$, the sender encodes t into a dense representation $h_0 = f^S(i_s)$ of size d. Then it iteratively samples symbols s_l based on the output $h_l = h^S\big(g(s_{l-1}), h_{l-1}\big)$ of the decoder (i.e. $\log \pi_l = h_l$ are unnormalized log-probabilities), producing the message $m = (s_1, \cdots, s_L)$.[4]

Based on the message m, the receiver computes dense representations $v = (v_i)_{1 \leqslant i \leqslant |C|}$, where $v_i = f^R(c_i)$ for each candidate object $c_i \in i_r$, and embeds m into $z = h_{l_0}$, where l_0 is the position of the first EOS symbol and $h_l = h^S\big(g^S(s_{l_0}), h_{l-1}\big)$.[5] Finally, the receiver predicts the target $t' \in \mathcal{C}$ by selecting the object corresponding to the highest probability of the Gibbs distribution computed as the dot product between z and each v_i. Agents' communication is successful provided that $t' = t$.

For disentangled input, the encoder $f(\cdot)$ is a single fully-connected layer; in the pixel input setting, we use the model described in Denamganaï et al. (2023), consisting of four 3×3 CNN layers with stride 2,[6], each of which is followed by a 2D batch normalization layer. The output of the last layer of size 1024 is transformed to a dense representation

[2] We do not remove homonym concepts.

[3] For each target vector, we sample 256/12/3 sets of distractors for training, ID and OOD evaluation. Note that the samples presented during training and during ID evaluation are independently drawn from the same distribution.

[4] At $l = 1$, a learned SOS embedding e_0 is used as the first argument of h.

[5] $h_0 = 0$ is assumed.

[6] The first/last two layers have 32/64 filters, respectively.

of size 128 by a fully connected layer. Outputs of the fully connected layer and each batch normalization layer are passed through the ReLU activation function. The symbol encoder $g(\cdot)$ maps previously sampled symbols to a dense representation of size 16, while the decoder $h(\cdot)$, is implemented as a single-layer LSTM with 64 and 128 hidden units for the first and second experiment, respectively.

3.3 Learning

Learning is based on Gumbel-Softmax relaxation (GS). During training, at each timestep $l \leqslant L_{\max}$ the sender transforms the vector of (unnormalized) log-probabilities $\log \pi_l$ into a relaxed symbol sample $\mathbf{s}_l \sim \mathrm{GS}(\pi_l, \tau)$, defined as:

$$\mathrm{GS}(\pi_l, \tau)_i = \frac{\exp\big((\log p_i + G_i)/\tau\big)}{\sum_{j \in A} \exp\big((\log p_i + G_j)/\tau\big)}$$

where $\pi_l = (p_i)_{i \in A}$ is the vector of (normalized) symbol probabilities, τ is the temperature parameter and each $G_j \sim \mathrm{Gumbel}(0, 1)$ (Huijben et al., 2023).

Unlike the majority of recent work on emergent communication systems, in which the whole setup is made deterministic on inference by applying argmax to the learned symbol distribution $\mathrm{Cat}(\pi)$, we use the same symbol sampling mechanism during training and evaluation. We take advantage of the fact that $\mathrm{argmax}_{i \in A}\big(\mathrm{GS}(\pi, \tau)\big) \sim \mathrm{Cat}(\pi, \tau)$,[7] to obtain discrete messages after the training is completed.[8] However, any feedback that impacts the training is exclusively based on relaxed symbol distributions, ensuring that gradients are not biased, as in the case of Straight-Through GS.[9]

While most research in EC minimizes cross-entropy loss of receiver output and target label, Kuciński et al. (2021) minimize cross-entropy between the attributes of the target and selected candidate instead. After preliminary experiments with both options (summarized in Appendix B), we opted for balancing these objectives by computing both losses:

$$\mathcal{L}_{\text{features}} = \sum_i H_c(\mathbb{1}_{A_i}, \tilde{A}_i)$$

$$\mathcal{L}_{\text{label}} = H_c(\mathbb{1}_{A_t}, \pi^R)$$

where for each attribute in (shape, color, x, y), indexed by i and taking the value A_i for the target t, the expected value $\tilde{A}_i$ of the selected object is a vector given by $\tilde{A}_i = \sum_{c_j \in \mathcal{C}} p_j^{\mathrm{R}} \mathbb{1}_{A_i^c}$ where p_j^{R} is the j-th position of the distribution π^{R} outputted by the receiver, representing the estimated probability that $t = c_j$.[10] We combine the losses as:

$$\mathcal{L} = \mathcal{L}_{\text{features}} + 0.5 \times \mathcal{L}_{\text{label}} + 0.01 \times \mathbb{E}(L)$$

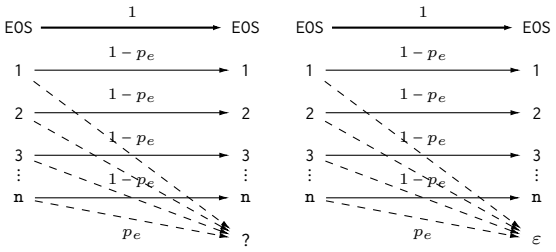
where the final term $\mathbb{E}(L)$ adds the length cost (Chaabouni et al., 2019a). Further details of our training procedure can be found in Appendix B.

3.4 Noisy Communication Channels

We consider two noise mechanisms, inspired by the erasure and deletion channels studied by information theory (Shannon, 1948; Mitzenmacher, 2009):

- Erasure channel: a non-EOS symbol is replaced by a special symbol, distinct from the symbols available to the sender.
- Deletion channel: a non-EOS symbol is removed from the message without replacement (an additional EOS symbol is appended).

Note that symbol positions and message length may only be distorted by the deletion channel, rendering the communicative task more difficult (*a priori*) given equal error probability as information on which symbols were distorted is lost.



(a) Erasure channel (b) Deletion channel

Figure 2: Models of the two noisy channels.

Figure 2 illustrates the behavior of the channels on transmission of a single symbol. Unlike other

[7] $\mathrm{Cat}(\pi, \tau)$ denotes temperature-adjusted categorical distribution based on π and can be computed as $\mathrm{softmax}(\pi, \tau)$. For brevity, we will henceforth implicitly assume that π is adjusted for the temperature.

[8] We additionally remove symbols preceded by EOS during training.

[9] The gradient is biased with respect to discrete symbol distributions.

[10] For a categorical RV X, $\mathbb{1}_X$ is its one-hot encoding.

work investigating noise in a differentiable setting (Foerster et al., 2016; Kuciński et al., 2021), we apply noise *after* sampling symbols. First, relaxed symbol representations are sampled according to the probability of error p_e (on evaluation, only non-EOS symbols are considered). Then, noise is applied to each symbol vector s_l: on training, non-EOS probability mass of the relaxed target symbol probabilities $s_l = \mathrm{GS}(\pi_l, \tau)$ is adjusted to represent the distribution after applying noise $C(s_l)$.

On evaluation, each one-hot vector s_l representing a non-EOS symbol is disrupted with probability p_e. We adjust the symbol distribution π of each message m passing through the channel C, to obtain symbol distribution $\mathbb{E}_{p_e}[C_{p_e}(\pi_l)] = p_e \times C(\pi_l) + (1 - p_e) \times \pi$ of the noisy message (analogously to adjusting relaxed symbol probabilities).[11] Efficiently solving the task requires that relevant

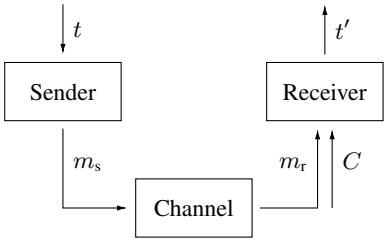


Figure 3: Standard information-theoretic communication scheme.

information about the target object be compressed into messages, which should also offer a degree of robustness to noise. The whole setup could be seen as a variant of the standard information-theoretic communication scheme (see Figure 3), in which the agents realize both source and channel coding of the target features, corresponding to these two goals. The sender encodes the features of the target object t into the message m_s; the receiver predicts the target object $t' \in C$ based on the message m_r after passing through the channel and additionally provided set of candidate objects C.

3.5 Information-Theoretic Measures

To formalize the intuition that agents' communication relies both on efficient encoding of the features of the target object (source coding) and on robustness against noise introduced by the channel (channel coding), let $M_\mathrm{S}, M_\mathrm{R}$ be RVs representing sent and received messages and T be a (uniformly distributed) RV representing disentangled features of the target object. Note that successful identification of the target object by the receiver depends on $I(T; M_\mathrm{R})$, which is bounded by both $I(T; M_\mathrm{S})$ and $I(M_\mathrm{S}; M_\mathrm{R})$: since M_R and T are conditionally independent given M_S, we have $I(T; M_\mathrm{R} \mid M_\mathrm{S}) = 0$. Optimizing the efficiency of channel coding and source coding of the protocol corresponds to maximizing the former and the latter, respectively. We assess the relative importance of these objectives at different stages of training by computing the values of $I(M_\mathrm{S}; M_\mathrm{R}), I(T; M_\mathrm{S})$, and $I(T; M_\mathrm{R})$ after every epoch.

Entropy estimators based on empirical distributions are proved to be biased and do not scale well with the support size of the distribution (Paninski, 2003). [12] This problem constitutes a major obstacle in a deterministic setting, in which the number of unique messages is bounded by the number of unique target objects. We leverage the access to actual symbol distributions that the sender samples from to compute entropy directly from probabilities or assume the Monte Carlo approach, reducing the bias by sampling additional messages.[13] Unless otherwise stated, reported values of entropy and other information-theoretic measures based on a Monte Carlo sample are computed using a Maximum a posteriori estimator with a Dirichlet prior $\alpha = 1/K$, where K is the number of all possible messages (Wolpert and Wolf, 1995; Perks, 1947). The method of computing entropy directly from probabilities is described in Appendix C. Furthermore, in Appendix D we compare the performance of several entropy estimators used on a single message or multiple samples against entropy computed from probabilities.

4 Results

We report results after training for 77,184 steps in the Disentangled input setting and 34,560 steps for the Pixel input.[14] We run simulations for maximum lengths $L \in \{2, 3, 4, 5\}$ and vocabulary size $|A| = 10$.[15]

We evaluate messages before and after passing

[11] In case of the deletion channel, we adjust relaxed probabilities by: (i) computing expected non-EOS symbol distributions at each position, and (ii) adjusting EOS probabilities at each position accordingly.

[12] In our case, there are $\mathcal{O}(|A|^{L_\mathrm{max}})$ possible messages. Even though we assume a moderate value of $|A| = 10$, the number of possible messages may easily exceed the number of unique sender inputs on evaluation.

[13] Neither of these approaches would work if argmax was applied, as described in subsection 3.3.

[14] The number of training steps was selected to ensure convergence, based on the preliminary experiments.

[15] Excluding the additional symbol for the erasure channel.

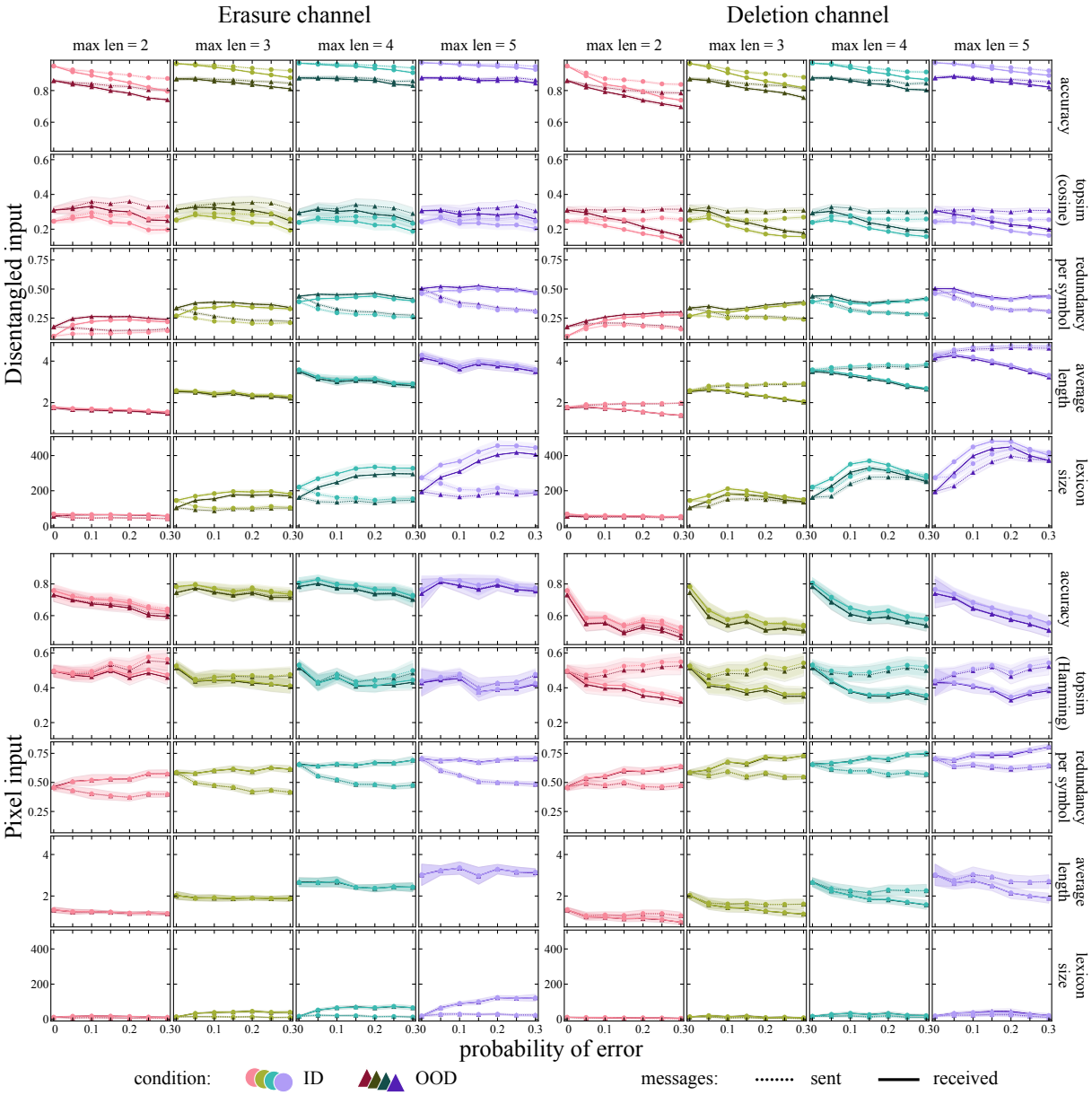


Figure 4: Accuracy, topographic similarity, redundancy per symbol, average message length and lexicon size, averaged over 20 trained models. Shaded areas represent 95% CIs.

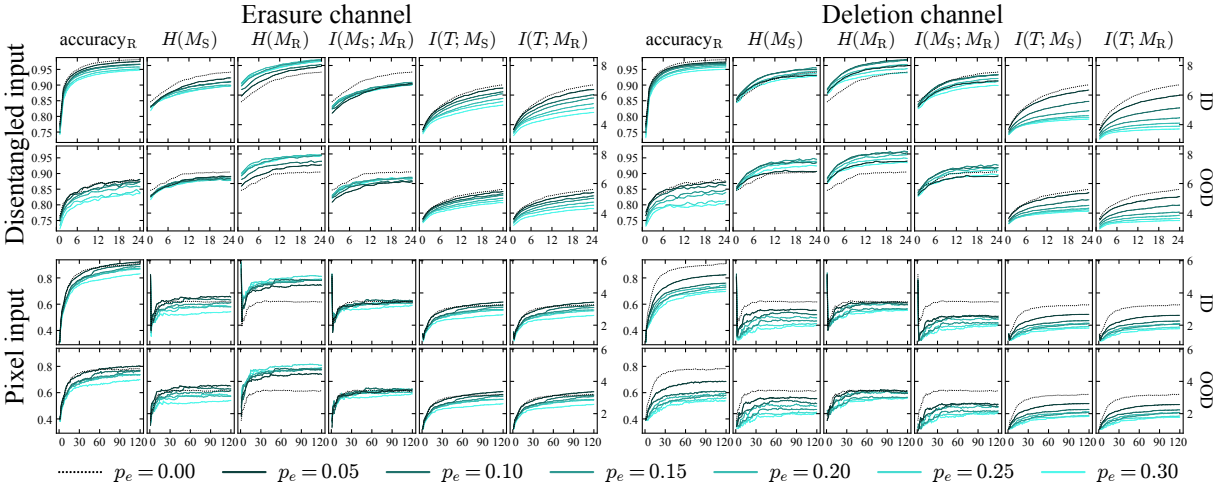


Figure 5: Average entropy (H) and mutual information (I) during training, for maximum length 4.

through the channel, henceforth referring to them as *sent* and *received* messages, correspondingly.[16] We analyze the resilience of the protocols to other mechanisms of message disruption and provide additional analysis to support our findings in section E.

Efficiency of communication As illustrated in Figure 4, in nearly all settings the success rate tends to decrease as more noise is introduced. In the disentangled input condition, we observe a consistent drop in success rate of around 10 pp. in the OOD condition relative to the ID condition. No such discrepancy is observed in experiments based on pixel input. Average accuracy in the pixel input experiments is substantially lower than for the disentangled input runs, although it is well above chance level (20%) for every condition. Simulations with larger maximum lengths result in more accurate communication. Notably, performance for the sent and received messages is almost the same after training on pixel input. Unsurprisingly, the disparity between success rate for sent and received messages is strongest for shorter messages.

Source vs Channel coding In a noisy environment, communicative efficiency has different components: to what extent do agents prioritize maximizing the informativeness of messages with respect to the target referent (source), versus enhancing robustness against channel noise. The results of this analysis are presented in Figure 5, for $L_{\max} = 4$ (see also Figure 13 in Appendix E for all maximum lengths).

For both channels and both input data types, we observe that the values $I(T; M_{\mathrm{S}})$, and $I(M_{\mathrm{S}}; M_{\mathrm{R}})$ consistently increase during training (with a steeper increase for lower noise levels), suggesting that the two objectives are equally important. On the test set for the disentangled input data, any amount of noise for either of the channels negatively impacts accuracy throughout training. This effect is also observed in the pixel input condition for the deletion channel; however, for the erasure channel, low and moderate noise levels also achieve a success rate comparable to the condition without noise (particularly in the final phases of training).

In case of the erasure channel we observe similar dynamics of change across training for both input data types: noise seems to positively impact the values of $H(M_{\mathrm{R}})$, whereas $H(M_{\mathrm{S}})$, $I(T; M_{\mathrm{S}})$ and $I(T; M_{\mathrm{R}})$ are consistently impeded proportionally to p_e. For the deletion channel, the impact of noise on $I(T; M_{\mathrm{S}})$ and $I(T; M_{\mathrm{R}})$ is even stronger, however the pattern of change of $I(M_{\mathrm{S}}; M_{\mathrm{R}})$ and entropy values is different for each input data type: the values of $I(M_{\mathrm{S}}; M_{\mathrm{R}})$, $H(M_{\mathrm{S}})$ and $H(M_{\mathrm{S}})$ all positively related to p_e on the disentangled dataset, but negatively related on the pixel input dataset.

Overall, we find that $I(M_{\mathrm{S}}; M_{\mathrm{R}})$, $H(M_{\mathrm{R}})$ and $H(M_{\mathrm{S}})$ are non-decreasing during training (except for the $p_e = 0$ condition where $I(M_{\mathrm{S}}; M_{\mathrm{R}}) = H(M)$). Although $I(T; M_{\mathrm{R}})$ is successfully optimized during training, the success rate does not reach values close to 1 in the pixel input condition. Concurrently, we find that in the pixel input setting the values of $I(T'; M_{\mathrm{R}})$, where T' represents the attributes of the object selected by the receiver, are much closer to $I(T; M_{\mathrm{R}})$ than on the disentangled input, both for training and evaluation (see Figure 14 in section E).

Lexical Properties We find that both types of noisy channels have a clear impact on properties of the emerging communication protocols. The average message length is affected in different ways (see *average length* row on Figure 4): (i) we observe a weak but consistent negative relationship between message length and p_e for the erasure channel; (ii) in case of the deletion channel and the disentangled input, the presence of noise increases the average length of sent messages, resulting in a values almost equal to $L_{\max}$, whereas average length of received messages is highest for $p_e = 0.05$ and steadily decreases for $p_e > 0.05$; in the pixel input condition, average length consistently decreases relative to p_e and in case of the received messages and is non-increasing for the sent messages. For the erasure channel, while the number of unique received messages mostly increases with noise, the number of unique sent messages remains stable. For the deletion channel, lexicon size for the received messages takes an inverted U shape and tends to increase with noise for sent messages in the disentangled condition. In general, lexicon size is much lower in the pixel setting, likely due to a lower number of unique target objects or the uniform distribution of target features. Lastly, redundancy per symbol and maximum message length seem to be positively related because permitting longer messages leaves more room for redundancy.

[16] Agents' weights are updated based solely on the received messages.

Redundancy We evaluate message redundancy based on the standard information-theoretic notion of relative redundancy, computed as $R(M) = 1 - {}^{H(M)}/_{H_{\max}(M)}$, where $H_{\max}(M)$ is the maximum achievable entropy given the expected length distribution of messages M.[17] The resulting value corresponds to redundancy per symbol, rather than redundancy of the whole sequence.

Across all conditions, we observe that the redundancy of the protocols in the ID and OOD conditions tend to coincide (see *redundancy per symbol* row in Figure 4). The presence of noise stimulates redundancy of the received messages; however, this is not the case for the sent messages: contrary to our expectations, we find that redundancy of sent messages tends to decrease relative to p_e in most cases, possibly because average message length decreases as well.[18]. Similar redundancy values were observed in additional experiments where the length-cost term in the loss function was disabled (see Figure 16 in section E), suggesting that this cost did not contribute to the protocol's lack of redundancy. For disentangled input and the erasure channel (or sufficiently long messages and the deletion channel), redundancy of noisy messages peaks for intermediate noise levels.

Compositionality We compute topographic similarity (Brighton and Kirby, 2006) to assess compositionality of the emerged protocols (see *topsim* in Figure 4). We opt for *topsim* due to its wide acceptance and its flexibility with regards to message length (it is unclear whether other compositionality metrics in the literature are applicable to variable-length sequences). *Topsim* requires the use of two distances: for the messages, we use Levenshtein, and for the targets, we use cosine for disentangled input and Hamming for pixel input.

We find that the erasure channel positively impacts topsim of sent messages when using moderate amounts of noise, provided the input is disentangled –the pattern is much less clear for the pixel input. The impact of the input representation is reversed for the deletion channel: while noise has an almost imperceptible effect for disentangled input, intermediate and high levels of noise positively impact topsim of the sent messages.

As expected, the *received* messages lose some of their compositional structure after passing through the noisy channels. This reduction of topsim largely follows the same relation to p_e in the Erasure channel, but the decrease is much more abrupt in the Deletion channel. This reduction is consistent with *average length*: the compositional structure is lost as messages become shorter.

The maximum sequence length has a small positive influence on the disentangled input, yet a (smaller) negative effect on the pixel input. We do not observe a relationship between *topsim* and *accuracy* within simulations (with the exception of a negative correlation for the deletion channel and received messages). The discrepancy between the sent and received conditions is much stronger for *topsim* than for *accuracy*.

5 Discussion

Our results suggest that, in some settings, agents opt for increasing compositionality as a way to combat noise. This is in line with earlier work (Nowak and Krakauer, 1999; Kuciński et al., 2021); however, here we show that this is also the case for erasure and deletion channels with variable-length messages; furthermore, the gains in compositionality largely depend on the type of noise and whether the input is disentangled.

This strategy followed by the agents has also been attested in studies that investigate the influence of population size in emergent languages. Larger groups inevitably introduce variability in the system, and studies with human subjects find that speakers tend to develop more structured languages in such situations, likely to preserve mutual understanding (Raviv et al., 2019). These findings have been replicated in a heterogeneous population of neural agents (Rita et al., 2021).

Consistent with Chaabouni et al. (2020), we do not find a direct relation between accuracy and compositionality; however, in our case, this may be due to the presence of noise, since more compositional systems are also those with higher probability of message distortion.

Our simulations use a different type of noise compared to previous approaches: instead of introducing other symbols from the vocabulary (Cope and Schoots, 2024; Kuciński et al., 2021) (which possibly have an attributed meaning), it distorts the

[17] Expected length is computed via conditioning on prefix probabilities (see Appendix C). $H_{\max}(M)$ for received messages computed based on the value of p_e, with the assumption that sending messages consisting of uniformly distributed (and so pairwise independent) non-EOS symbols leads to maximal entropy of noisy messages.

[18] Increasing message length while holding entropy constant, e.g. by repeating symbols, would result in higher redundancy, since $H_{\max}$ positively depends on actual message length.

message by either introducing an out-of-vocabulary symbol (*erasure*) or removing part of the message (*deletion*). While the erasure is arguably easier to combat, the latter may increase the difficulty, especially if agents rely on positional information to interpret meaning. Interestingly, the agents seem to deal with both types of noise equally well in the disentangled input condition, but in the case of pixel input, accuracy is substantially lower for the deletion channel. Concurrently, redundancy rate and average length of the sent messages remain stable relative to the noise level, suggesting that in the pixel input setting, the sender counters deletion by generating more compositional messages rather than by increasing redundancy (either by producing longer messages or increasing redundancy per symbol).

Vital et al. (2025) find that protocols emerging in environments in which symbols may be erased are more resilient to symbol masking, and argue that this results from a higher degree of redundancy. While we observe a similar effect (see section E), redundancy at the symbol level of messages generated by the sender decreases or does not change with respect to the noise level (in case of the deletion channel and disentangled input condition, lower redundancy per symbol is compensated by increased message length). These results indicate that the increased robustness of the protocols emerging in noisy environments may be accounted for by compositionality or other features, rather than by redundancy.

In our setting, agents perform joint source-channel coding with constrained maximum message length. Xin et al. (2024) note that in scenarios where the coding complexity is limited, the joint approach surpasses the traditional two-step approach. We observe that $I(M_S; T)$ $I(M_S; M_R)$ both increase during training, and for a given value of p_e, the rate of change for these two values is proportional. This indicates that the objectives of conveying the information about the target in the message and maximizing noise-resilience of the protocol both guide the training. Furthermore, Gündüz et al. (2024) argue that grateful degradation with noise is an advantage of joint source-channel coding; indeed, we observe that the rate at which the aforementioned values and $I(M_R; T)$ increase, gradually decreases as we introduce more noise.

6 Conclusions

We set out to investigate the influence of two types of channel noise (erasure and deletion) in EC with variable message length. To this end, we run simulations of the Lewis referential game, using the setup of Lazaridou et al. (2018) with the following modifications: (i) including two noisy channels, (ii) using Gumbel-Softmax relaxation, (iii) sampling symbols according to learned distributions, instead of evaluating in a deterministic setting. Our work provides evidence that this type of noise provides a bias favorable to compositionality also in systems that allow for variable message length, but the choice of specific noise type and probability, as well as input representation, matters.

Limitations

On training, our design of noisy communication channels relies on modifying relaxed symbol vectors. While our design of the channels aims to ensure that the mechanism is consistent between relaxed and discrete symbols, i.e. during training and evaluation, other implementations are possible. For instance, consider the following approaches to erasing a single relaxed symbol vector, i.e. replacing the symbol with the special ? symbol:

(i) assigning a part of the non-EOS probability mass given by p_e of every symbol passed through the channel to ?, without sampling the symbols to be disrupted;
(ii) sampling symbols to be disrupted according to p_e, for which the whole non-EOS probability mass is assigned to ? (our implementation);
(iii) sampling specific positions of each relaxed symbol representation to be disrupted according to p_e, in which case a symbol vector could consist of both disrupted and disrupted probabilities after passing through the channel.

All of the above implementations could be argued to be consistent with our interpretation of symbol erasure, yet they could impact communication in different ways.[19]

We acknowledge that some regularities specific to natural language may possibly require a joint presence of several naturalistic pressures. Several adjustments of the EC simulation setup have

[19]Note that this would not be a problem if discrete symbols were used on training, e.g. for learning based on REINFORCE.

been argued to be of crucial importance for modeling naturalistic communication pressures, some of which include periodical parameter resetting, increasing the population size of agents, switching agent roles, voting, and imitation among sender agents (Chaabouni et al., 2022; Galke and Raviv, 2025). Furthermore, Chaabouni et al. (2022) emphasizes the significance of considering complex tasks. The incorporation of the aforementioned mechanisms in our setup could potentially reveal additional effects.

Our analysis of compositionality is entirely based on topographic similarity, which has been argued to have multiple drawbacks (Korbak et al., 2020; Chaabouni et al., 2020). Alternative measures include tree reconstruction error (Andreas, 2019), conflict count (Kuciński et al., 2021), as well as positional disentanglement and bag-of-symbols disentanglement (Chaabouni et al., 2020). Notwithstanding, (to the best of our knowledge) there is no consensus as to which method best captures compositionality (including its subtler forms), especially in a variable message length setting.

Our accuracy for the pixel input in the no noise condition is lower compared to previous work (e.g. Lazaridou et al., 2018; Kuciński et al., 2021) report an accuracy of around 90% or higher for pixel input, compared to around 80% for sufficiently long message lengths). This might be due to the stochastic evaluation setting and the temperature value used ($\tau = 1$ after training). [20] We also note several differences between input data, architectures, and learning mechanisms compared to the aforementioned work.

Acknowledgments

This work is en extension of a final project for the Advanced Neural and Cognitive Modeling course at the University of Amsterdam and was completed by C.K. and F.K. as an individual project under the supervision of R.G.A., as part of the Master of Logic program. R.G.A. was financed by the NWO SSH Open Competition XS grant (project no. 406.XS.24.01.128).

[20] Kuciński et al. (2021) assumes a constant number of classes for each attribute, making the task simpler. Guided by the intuition that the different number of classes should be balanced out, we experimented with adjusting the logarithm base used to compute the entropy of each attribute (assuming the number of possible values as the base); however, this lead to poorer performance.

References

Daniel Akkerman, Phong Le, and Raquel G. Alhama. 2024. The emergence of compositional languages in multi-entity referential games: from image to graph representations. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 18713–18723, Miami, Florida, USA. Association for Computational Linguistics.

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

Matthew Aylett. 2000. Stochastic suprasegmentals: Relationships between redundancy, prosodic structure and care of articulation in spontaneous speech. In *Sixth International Conference on Spoken Language Processing*, pages 646–649. ISCA.

Carla Bazzanella. 2011. Redundancy, repetition, and intensity in discourse. *Language sciences*, 33(2):243–254.

Barend Beekhuizen, Rens Bod, and Willem Zuidema. 2013. Three design principles of language: The search for parsimony in redundancy. *Language and speech*, 56(3):265–290.

Henry Brighton and Simon Kirby. 2006. Understanding linguistic evolution by visualizing the emergence of topographic mappings. *Artificial 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.

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

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.

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*.

Marwa Chafii, Salmane Naoumi, Reda Alami, Ebtesam Almazrouei, Mehdi Bennis, and Merouane Debbah. 2023. Emergent communication in multi-agent reinforcement learning for future wireless networks. *IEEE Internet of Things Magazine*, 6(4):18–24.

Edward Choi, Angeliki Lazaridou, and Nando de Freitas. 2018. Compositional obverter communication learning from raw visual input. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*.

Dylan Cope and Nandi Schoots. 2024. Channel randomisation methods for zero-shot communication. In *Proceedings of the European Conference on Artificial Intelligence, ECAI-24*, pages 3620–3627. IOS Press. Main Track.

Gavin E Crooks. 2024. On measures of entropy and information. Accessed: 2025-03-05.

Kevin Denamganaï, Sondess Missaoui, and James Alfred Walker. 2023. Visual referential games further the emergence of disentangled representations. *arXiv preprint arXiv:2304.14511*.

Jakob Foerster, Ioannis Alexandros Assael, Nando de Freitas, and Shimon Whiteson. 2016. Learning to communicate with deep multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.

Lukas Galke, Yoav Ram, and Limor Raviv. 2022. Emergent communication for understanding human language evolution: What's missing? In *Emergent Communication Workshop at ICLR 2022*.

Lukas Paul Achatius Galke and Limor Raviv. 2025. Learning and communication pressures in neural networks: Lessons from emergent communication. *Language Development Research*, 5(1):116–143.

Abhinav Gupta, Cinjon Resnick, Jakob Foerster, Andrew Dai, and Kyunghyun Cho. 2020. Compositionality and capacity in emergent languages. In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 34–38, Online. Association for Computational Linguistics.

Deniz Gündüz, Michèle A. Wigger, Tze-Yang Tung, Ping Zhang, and Yong Xiao. 2024. Joint source–channel coding: Fundamentals and recent progress in practical designs. *Proceedings of the IEEE*, pages 1–32.

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.

Iris A. M. Huijben, Wouter Kool, Max B. Paulus, and Ruud J. G. van Sloun. 2023. A review of the gumbel-max trick and its extensions for discrete stochasticity in machine learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):1353–1371.

Patrick Juola. 1998. Cross-entropy and linguistic typology. In *New Methods in Language Processing and Computational Natural Language Learning*.

Eugene Kharitonov, Rahma Chaabouni, Diane Bouchacourt, and Marco Baroni. 2019. EGG: a toolkit for research on Emergence of lanGuage in Games. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*, pages 55–60, Hong Kong, China. Association for Computational Linguistics.

Eugene Kharitonov, Rahma Chaabouni, Diane Bouchacourt, and Marco Baroni. 2020. Entropy minimization in emergent languages. In *International Conference on Machine Learning*, pages 5220–5230. PMLR.

Anastasios Kontogiorgis and Mélanie Bouroche. 2024. CAV unsignalised intersection crossing with unreliable emergent communication. In *Thirteenth International Workshop on Agents in Traffic and Transportation co-located with the the 27th European Conference on Artificial Intelligence (ECAI 2024), Santiago de Compostela, Spain, October 19, 2024*, volume 3813 of *CEUR Workshop Proceedings*, pages 54–68. CEUR-WS.org.

Tomasz Korbak, Julian Zubek, and Joanna Rączaszek-Leonardi. 2020. Measuring non-trivial compositionality in emergent communication. *Preprint*, arXiv:2010.15058.

Tom Kouwenhoven, Max Peeperkorn, Bram Van Dijk, and Tessa Verhoef. 2024. The curious case of representational alignment: Unravelling visio-linguistic tasks in emergent communication. *arXiv preprint arXiv:2407.17960*.

Łukasz Kuciński, Paweł Kołodziej, and Piotr Miłoś. 2020. Emergence of compositional language in communication through noisy channel. In *Language in Reinforcement Learning Workshop at ICML 2020*.

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*, 34:23075–23088.

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

Angeliki Lazaridou, Karl Moritz Hermann, Karl Tuyls, and Stephen Clark. 2018. Emergence of linguistic communication from referential games with symbolic and pixel input. *arXiv preprint arXiv:1804.03984*.

Nunzio A. Letizia, Andrea M. Tonello, and H. Vincent Poor. 2023. Cooperative channel capacity learning. *IEEE Communications Letters*, 27(8):1984–1988.

David Kellogg Lewis. 1969. *Convention: A Philosophical Study*. Wiley-Blackwell, Cambridge, MA, USA.

Yuchen Lian, Arianna Bisazza, and Tessa Verhoef. 2021. The effect of efficient messaging and input variability on neural-agent iterated language learning. *arXiv preprint arXiv:2104.07637*.

Michael Mitzenmacher. 2009. A survey of results for deletion channels and related synchronization channels. *Probability Surveys*, 6(none):1 – 33.

Mitja Nikolaus. 2024. Emergent communication with conversational repair. In *The Twelfth International Conference on Learning Representations*.

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

Liam Paninski. 2003. Estimation of entropy and mutual information. *Neural Computation*, 15(6):1191–1253.

Wilfred Perks. 1947. Some observations on inverse probability including a new indifference rule. *Journal of the Institute of Actuaries*, 73(2):285–334.

Jannik Peters, Constantin Waubert de Puiseau, Hasan Tercan, Arya Gopikrishnan, Gustavo Adolpho Lucas de Carvalho, Christian Bitter, and Tobias Meisen. 2025. Emergent language: a survey and taxonomy. *Autonomous Agents and Multi-Agent Systems*, 39(1):18.

Yue Pi, Wang Zhang, Yong Zhang, Hairong Huang, Baoquan Rao, Yulong Ding, and Shuanghua Yang. 2024. Applications of multi-agent deep reinforcement learning communication in network management: A survey. *CoRR*, abs/2407.17030.

Limor Raviv, Antje Meyer, and Shiri Lev-Ari. 2019. Larger communities create more systematic languages. *Proceedings of the Royal Society B*, 286(1907):20191262.

Yi Ren, Shangmin Guo, Matthieu Labeau, Shay B Cohen, and Simon Kirby. 2020. Compositional languages emerge in a neural iterated learning model. *arXiv preprint arXiv:2002.01365*.

Mathieu Rita, Florian Strub, Jean-Bastien Grill, Olivier Pietquin, and Emmanuel Dupoux. 2021. On the role of population heterogeneity in emergent communication. In *International Conference on Learning Representations*.

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. *Advances in neural information processing systems*, 35:1389–1404.

Claude E Shannon. 1951. The redundancy of english. In *Cybernetics; Transactions of the 7th Conference, New York: Josiah Macy, Jr. Foundation*, pages 248–272.

Claude Elwood Shannon. 1948. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423.

Carina Silberer, Vittorio Ferrari, and Mirella Lapata. 2013. Models of semantic representation with visual attributes. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 572–582.

David Simões, Nuno Lau, and Luís Paulo Reis. 2019. Multi-agent deep reinforcement learning with emergent communication. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.

Agnieszka Słowik, Abhinav Gupta, William L Hamilton, Mateja Jamnik, Sean B Holden, and Christopher Pal. 2020. Structural inductive biases in emergent communication. *arXiv preprint arXiv:2002.01335*.

Zoltán Gendler Szabó. 2004. The compositionality papers. *Mind*, 113(450):340–344.

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*, pages 60–70, Online. Association for Computational Linguistics.

Fábio Vital, Alberto Sardinha, and Francisco S. Melo. 2025. Implicit repair with reinforcement learning in emergent communication. *Preprint*, arXiv:2502.12624.

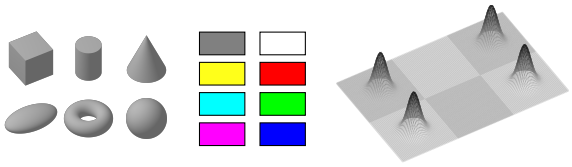
Jannis Weil, Gizem Ekinci, Heinz Koeppl, and Tobias Meuser. 2023. Learning to cooperate and communicate over imperfect channels. *Preprint*, arXiv:2311.14770.

David H. Wolpert and David R. Wolf. 1995. Estimating functions of probability distributions from a finite set of samples. *Phys. Rev. E*, 52:6841–6854.

Gangtao Xin, Pingyi Fan, and Khaled B. Letaief. 2024. Semantic communication: A survey of its theoretical development. *Entropy*, 26(2).

A Pixel Input Dataset Details

Scene generation For each combination of the 6 shapes, 8 colors, and positions on a 2×2 grid (see Figure 6), we generate 20 input images.[21] Example images are presented in Figure 7. Object rotation is uniformly sampled, and since most objects exhibit rotational symmetry, we do not consider it an attribute. Object location is uniformly sampled among 4 combinations of two positions on each axis, after which it is shifted by an offset sampled from a normal distribution, effectively resulting in the distribution depicted in Figure 6c. We aim to eliminate the possibility of basing the protocol on the color of the tile occupied by the target object, rather than its x and y coordinates. We do not allow horizontally centered object positions, so that the tile color coincides with the position on the y axis.



(a) 6 shapes (b) 8 colors (c) 4 positions on a grid

Figure 6: Object attributes for the Obverter dataset.

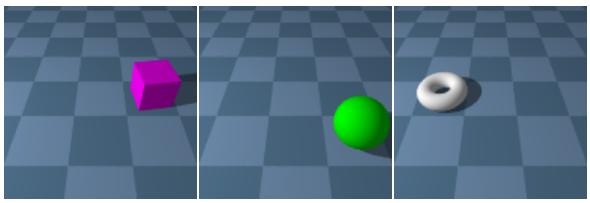


Figure 7: Example input images

Sampling distractors Combinations of shape and color are divided into exclusive train and test sets, as illustrated in Table 1. The train set is used for training and ID evaluation, whereas the test set is only used for OOD evaluation. Distractors are uniformly sampled from the available combinations of object shape, color, and position. Additionally, in 50% of the samples, it is ensured that each combination of object shape and color is unique within the sample (i.e. objects of the same shape and color but at different positions may not co-occur.)

[21] We utilize the code made available by Choi et al. (2018): https://github.com/benbogin/obverter/blob/master/create_ds.py

	white	gray	red	yellow	green	cyan	blue	magenta
cube	•						•	
sphere		•						•
cylinder	•		•					
torus		•		•				
ellipsoid			•		•			
cone				•		•		

Table 1: Division of combinations of shape and color into the train and test datasets. Marked combinations have been assigned to the test dataset.

B Training

Computational Resources The models used in the disentangled input experiments contained 119,162 parameters, while those used in the pixel input experiments comprised 545,082 parameters. Since running simulations on GPUs did not lead to considerable reduction of the training time, we opted for the cheaper CPU setup: all experiments were run on CPU-only computing nodes with 192 cores and 384 GB of RAM; up to 96 simulations were ran in parallel on each node. On average, running a single experiment took approximately 32 minutes in the disentangled setting or 5 hours and 11 minutes hours in the pixel setting.

Objective Function. Minimizing cross entropy between receiver output and label is a widely used training objective (Kharitonov et al., 2020), which ensures the identity of the target object is the only feedback signal guiding the training. Kuciński et al. (2021) argue that spontaneous emergence of compositional communication requires the presence of inductive biases, and opt to minimize cross entropy between the attributes of the target and selected object instead. Relying on the features of the target object makes it possible to reward partial success, since the penalty for selecting a distractor depends on how similar it is to the target object.

In preliminary experiments we found that relying solely on the label loss results in a slower convergence of the training and scarcely leads to emergence of compositional communication. On the other hand, while minimizing cross-entropy between the features of the target and selected objects guides the agents to develop both highly efficient and compositional protocols in some cases, we observe that the training is less stable, and in some

runs, the agents would fail to establish efficient protocols. The approach we report strikes a good balance.

Hyperparameters In all experiments, we assume a batch size of 32 and use the AdamW optimizer with weight decay 0.01. In the pixel input setting, weight decay is not applied to the CNN module. The learning rates used for the sender/receiver are 5e-3/1e-3 in the disentangled input condition or 5e-4/1e-4 on the pixel input dataset.

We anneal the GS temperature from 1.5 in the first epoch to 1 in the last epoch, according to an exponential decay schedule. Kharitonov et al. (2020) suggest that in GS, the pressure to minimize entropy of the protocol is stronger for lower temperature values; in our stochastic evaluation setting, increasing the degree of discreteness in the final phases of the training is especially desirable. Concurrently, higher temperature values may boost training, as they guarantee stronger gradients.

We find that in the pixel input condition, assuming a non-zero length cost sometimes impedes successful communication, preventing the speaker from generating any non-EOS symbols (after discretization). At the same time, applying length cost has been argued to be a prerequisite for the emergence of human-like communication protocols (Chaabouni et al., 2019a). We disable the length for the initial 250 network updates in the pixel input condition, which suffices to ensure that the training converges.

C Information-theoretic Measures

Computing Entropy Directly from Probabilities We compute message entropy M as the joint entropy of its symbols S_l, $M = (S_1, \cdots, S_L)$.[22] We do so by iteratively applying the chain rule:

$$H(M) = \sum_{l=1}^{L} H(S_l \mid S_{<l})$$

Computing $H(S_1)$ is straightforward, as it suffices to aggregate probabilities $(\pi_{i;l})_{1 \leqslant i \leqslant n} = \bar{\pi}_l$ corresponding to symbols $\bar{s}_1$.[23] Otherwise, if $l > 1$, we consider all possible prefixes $\tilde{s} = (\tilde{s}_j)_{j<l}$ such that each $\tilde{s}_j$ is (an index of) a non-EOS symbol. Next, we compute the probability $\pi_{i,\tilde{s}}$ of the prefix $\tilde{s}$ for every message m_i:

$$\pi_{i,\tilde{s}} = \prod_{j=1}^{l-1} \pi_{i;\tilde{s}_j}$$

Next, we aggregate symbols probabilities $\bar{s}_l$ weighted by the prefix probabilities $\pi_{i;\tilde{s}}$, as well as the prefix probabilities themselves between messages to obtain expected distribution $\pi^{\star}_{l|\tilde{s}}$ and prefix probabilities $\pi^{\star}_{\tilde{s}}$, respectively:

$$\pi^{\star}_{l|\bar{s}} = \sum_{i=1}^{n} \pi_{i,\tilde{s}}\, \pi_{i,l}$$

$$\pi^{\star}_{\tilde{s}} = \sum_{i=1}^{n} \pi_{i,\tilde{s}}$$

Finally, we compute $H(S_l|S_{<l})$ as follows:

$$H(S_l \mid S_{<l}) = \sum_{\tilde{s}} \pi^{\star}_{\tilde{s}}\, H(S_l \mid S_{<l} = \tilde{s})$$

$$= \sum_{\tilde{s}} \pi^{\star}_{\tilde{s}}\, H(\pi^{\star}_{l|\tilde{s}})$$

Each time we aggregate probabilities, we normalize them, which is not made explicit in the notation for the sake of clarity. When aggregating prefix probabilities, we normalize them relatively to the conditional non-EOS probability mass. All computations, except for the final step, are performed in log-space.

We use the above procedure to compute KL divergence reported in section E as well: given messages $M_i = (S_{i;j})_{0<j<k}$, $i \in \{1, 2\}$, it suffices to replace the entropy term in the final step with $D_{KL}\big((S_{1;l}|S_{1;<l}) \,||\, (S_{2;l}|S_{2;<l})\big)$.

Identifying communicated attributes In the pixel input experiment, we exploit the Monte Carlo approach to identify which object attributes serve as a warp for agents' communication. For each attribute in (shape, color, x, y), indexed by i, let A_i be its value for the target object. We compute mutual information for $I(M; A_i|A^t_{\neq i})$ between messages $\mathbf{M}$ and A_i given $A_{\neq i}$ where $A_{\neq i} = (A_j)_{j \neq i}$ is a compound RV taking the values of remaining attributes of the target. We then compute proficiency, also known as uncertainty coefficient, conditioned on $A_{\neq i}$, according to the formula:

$$U(A_i) = \frac{I(M; A_i|A_{\neq i})}{H(M, A_i|A_{\neq i})}$$

The measure represents the fraction of information contained in A_i that can be predicted based on M, given the remaining attributes (in order to exclude correlations between object attributes, e.g. during OOD evaluation, shape and color are

[22] $M, (S_l)_{l<L}$ are random variables whose realizations are messages $\bar{m} = \{m_i\}_{1 \leqslant i \leqslant n}$ and their symbols $(\bar{s}_l)_{l<L}$.

[23] All probabilities are temperature-adjusted, as described in subsection 3.3, which is not made explicit in the notation to prevent clutter.

strongly correlated); normalization ensures values within the $[0, 1]$ range. As shown in Figures 15a and 15b, agents tend to initially rely on object position, and the importance of object shape and color increases in the latter phases of the training, consistent with the results reported by Lazaridou et al. (2018), who found that the agents tend to base their communication on object location. Furthermore, we observe that the presence of noise has an impact on the communicated attributes: agents often more strongly rely on object location rather than shape or color for higher levels of noise applied during training.

D Benchmark of Entropy Estimation Methods

During training, we evaluate message entropy after every epoch, estimating it with 4 different estimators: maximum likelihood estimator (ML), James-Stein estimator, and maximum a posteriori estimator with a Dirichlet prior of $1/m$ (Perks) or $\sqrt{n}/m$ (Minimax), where m is the number of all permissible messages and n is the number of realizations. We use implementations from the `pyitlib` library and estimate entropy based on a single sample or after drawing multiple samples: 20/100 samples for the train/test dataset in the disentangled input condition or 40/200 samples in the pixel input condition. We then compute the bias of each estimate against the value computed directly from probabilities. The results for simulations with zero probability of error are presented in Figure 8. Simulations with non-zero probability of error were excluded to ensure a reliable comparison (e.g. in case of the erasure channel, the library does not permit specifying the known probability of the special ? symbol).

Although the James-Stein estimator proved most accurate overall, yielding more symmetric and marginally more closely centered around 0 distributions of bias compared to the other methods, the observed improvement over the simpler maximum likelihood estimator is almost negligible. However, drawing multiple samples (see subsection 3.5) consistently lead to markedly more accurate estimates.

Furthermore, we observe that as maximum message length increases, so does the degree to which the estimate is biased. While in the deterministic evaluation setting entropy estimated from a sample is deemed to be heavily biased for sufficiently long messages (assuming that the evaluation dataset has a limited size), our approach allows for estimating entropy based on an arbitrary number of samples, even for moderately sized datasets, resulting in more accurate estimations.

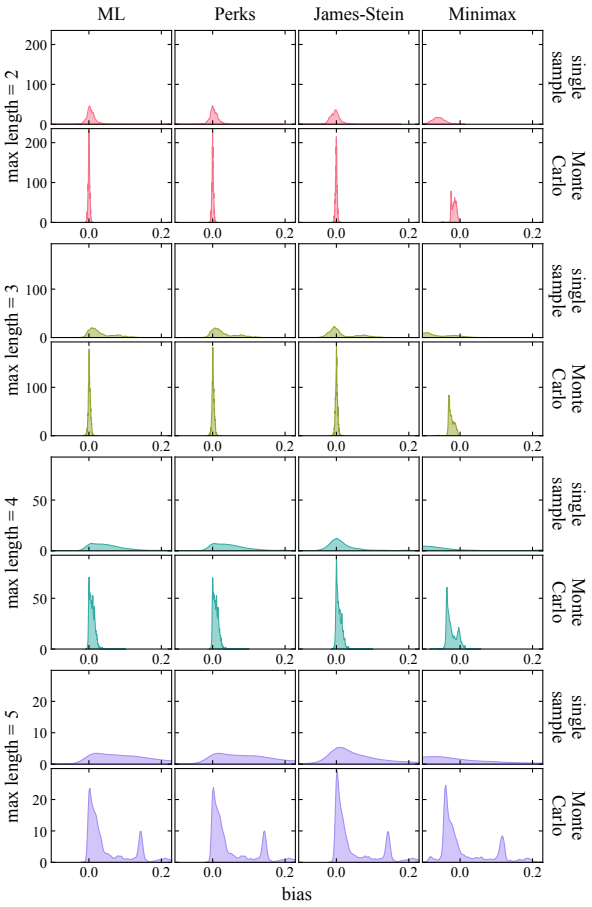


Figure 8: Distributions of the bias of entropy estimated from a sample of messages, computed for simulations with zero probability of error. Four entropy estimators are compared against entropy values computed from symbol distributions. Distributions of bias are based on message entropy estimates computed after every epoch, sourced from all simulations (values after the first two epochs were excluded). Probability density on the y axis was approximated using kernel density estimation.

E Extended Analyses

Average message length in the pixel setting In the pixel setting, average message length does not approach the maximum length as more noise is introduced. A possible explanation is that the alphabet size and maximum lengths permit communicating more information than necessary to solve the task: although assuming maximum length 3 would be sufficient to map a unique sequence to each possible combination of the 4 attributes,[24] the effect is observed also for maximum length 2. In spite of this, performance is far from perfect, since average success rate in the pixel setting does not exceed 85%.

Novelty of the protocol during generalization Estimating cross-entropy or relative entropy (KLD) was proposed as a method for quantifying the degree of similarity between natural languages (Juola, 1998). Since the chain rule for entropy resembles its counterpart for KLD (Crooks, 2024), the procedure described in subsection 3.5 can easily be adjusted to compute relative joint entropy between symbols of different distributions of sequences of symbols, making it possible to compute KLD directly from probabilities. We report KLD between messages in the ID and OOD conditions as a measure of similarity between the protocol used when agents communicate about known concepts and when they generalize.[25]

Figure 9 illustrates the relationship between KLD values and probability of error. We observe that the extent to which the protocols used to refer to objects in the ID and OOD conditions differ depends on the input data type used, rather than on the channel type. When training on the disentangled input, KLD values decrease as p_e increases, both for sent and received messages and consistently across all considered message lengths. These results suggest that the presence of noise may encourage the agents to exploit established linguistic patterns during generalization. On the other hand, we do not observe any strong relationship between KLD values and intensity of noise in the pixel input setting, indicating that this effect is based on naturalistic attribute co-occurrence patterns present only in the disentangled input setting. Although KLD monotonously decreases for the deletion channel, all values fall into the narrow $[2.5, 3.5]$ range, whereas the spread is much broader on the disentangled input.

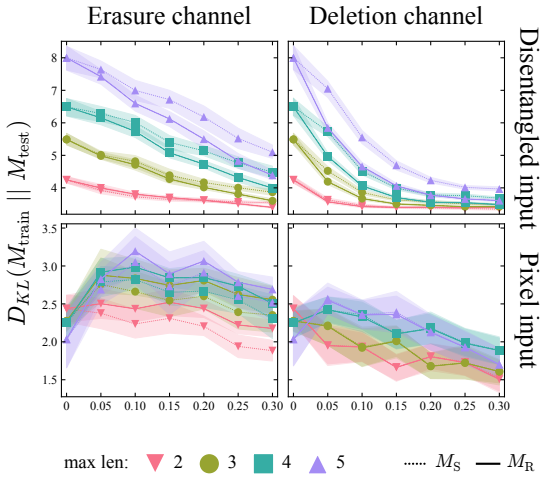


Figure 9: Average KLD between messages in the ID and OOD conditions for each input data type and maximum length, stratified by the probability of error. Shaded areas represent 95% CIs.

Resilience to Message Disruption Following Vital et al. (2025), we investigate whether communication protocols emerging in noisy environments are more robust to various types of message disruption. We consider the following perturbations:

- *deletion* of a single symbol (as in the case of the deletion channel),
- *replacement* of a single non-EOS symbol with a different non-EOS symbol (uniformly sampled, the replacement symbol may not be the special erased symbol),
- *permutation* of 2, 3, 4, or 5 non-EOS symbols (or all available symbols for shorter messages).

After training, we randomly sample 5 target non-EOS symbols to be disrupted (or combinations of symbols in case of permutation). We exclude permutations of n symbols in which some symbols do not change their position (i.e. permutations whose matrix representation has a non-zero diagonal). Finally, we pass the disrupted messages as input to the receiver. The difference between accuracy before and after perturbations is shown in Figure 12.

Unsurprisingly, we find that passing messages through the deletion channel during training results in protocols highly robust to this type of noise. For most settings and mechanisms of disruption, we

[24] 48 combinations of shape and color at 4 possible locations yield 192 combinations in total (144 for the objects in the train set only).

[25] We examine KLD in the other direction as well and find that its dependence on noise level follows a similar pattern.

observe that the presence of noise improves robustness of the protocols, since the observed difference between accuracy before and after disruption tends to be most prominent for $p_e = 0$. While the difference monotonously decreases relative to p_e for symbol permutations, the drop is often lowest for intermediate levels of noise for received messages or plateaus for sent messages. Notably, in case of the erasure channel the difference between accuracy before and after disruption does not seem to decrease as p_e increases (moreover, for sufficiently long messages and the disentangled input, we observe decreasing robustness to symbol deletion).

Additional figures The following figures illustrate our analyses:

- Figure 10: KLD between messages in the in-domain and out-of-domain conditions
- Figure 11: relationship between compositionality and generalization
- Figure 12: resilience to message disruption
- Figure 13: information-theoretic measures during training for all message lengths
- Figure 14: changes of entropy, topographic similarity and redundancy during training
- Figure 15: communicated attributes during training in the pixel input condition
- Figure 16: results for simulations with zero length cost

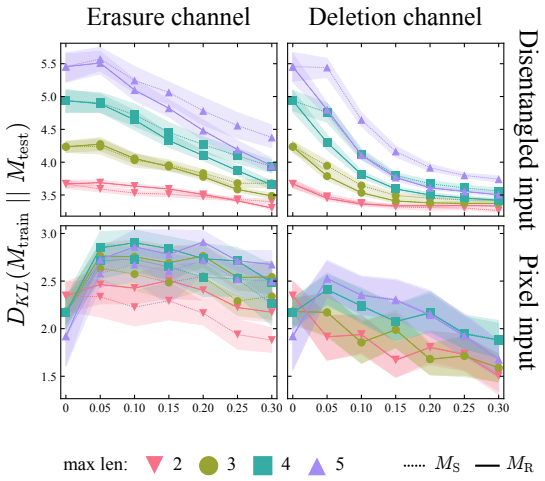


Figure 10: Average KLD between sent and received messages in the ID and OOD conditions for each input data type and maximum length, stratified by the probability of error. Shaded areas represent 95% CIs.

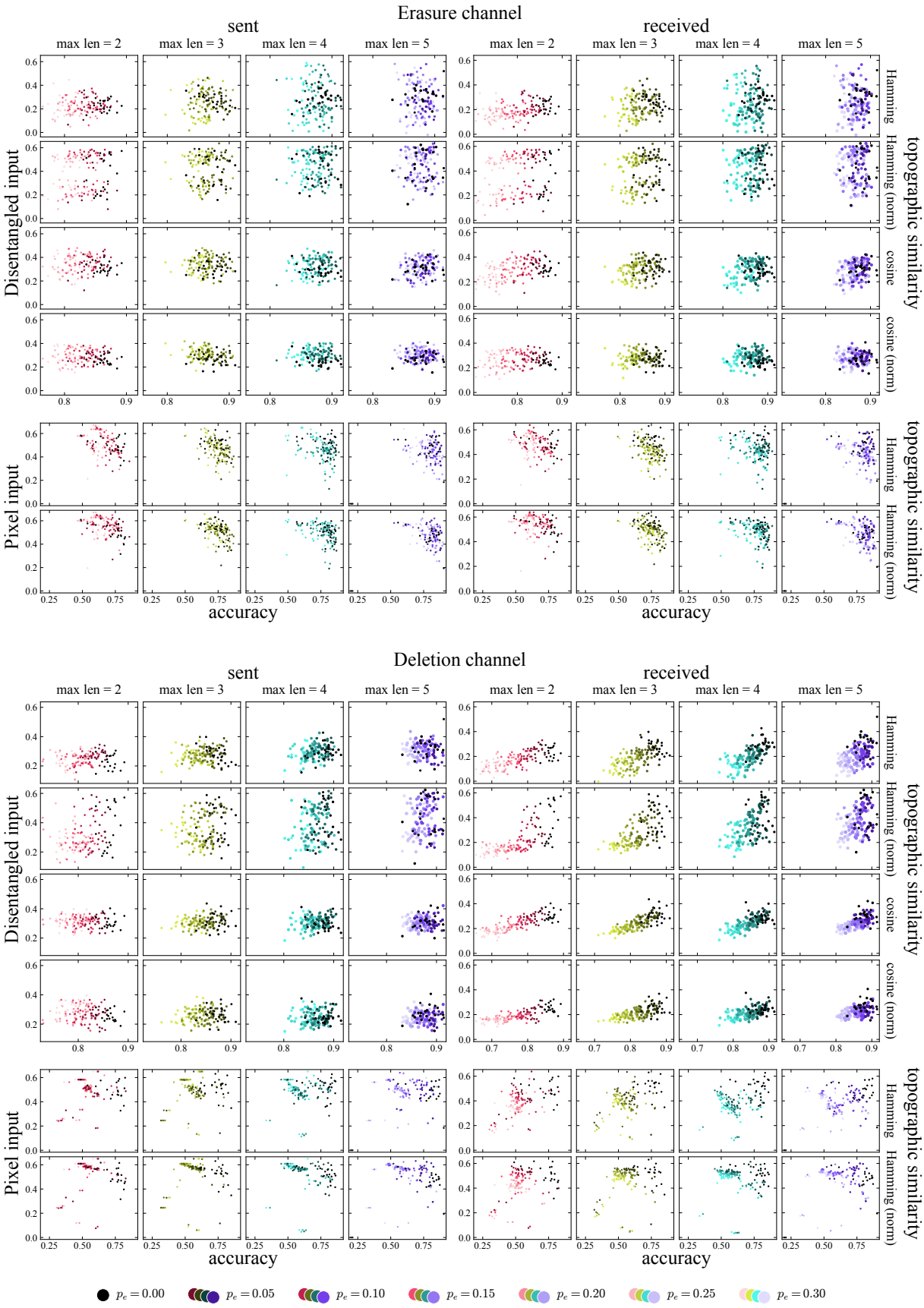


Figure 11: Accuracy, topographic similarity and lexicon size for final out-of-domain evaluation (i.e. after completing the training). Each point corresponds to a single run. Marker size represents lexicon size. Top. sim. was computed using Levenshtein edit distance in the message space (normalized with respect to average length of the two messages or not), and Hamming distance in the meaning space, as well as cosine distance on the disentangled input.

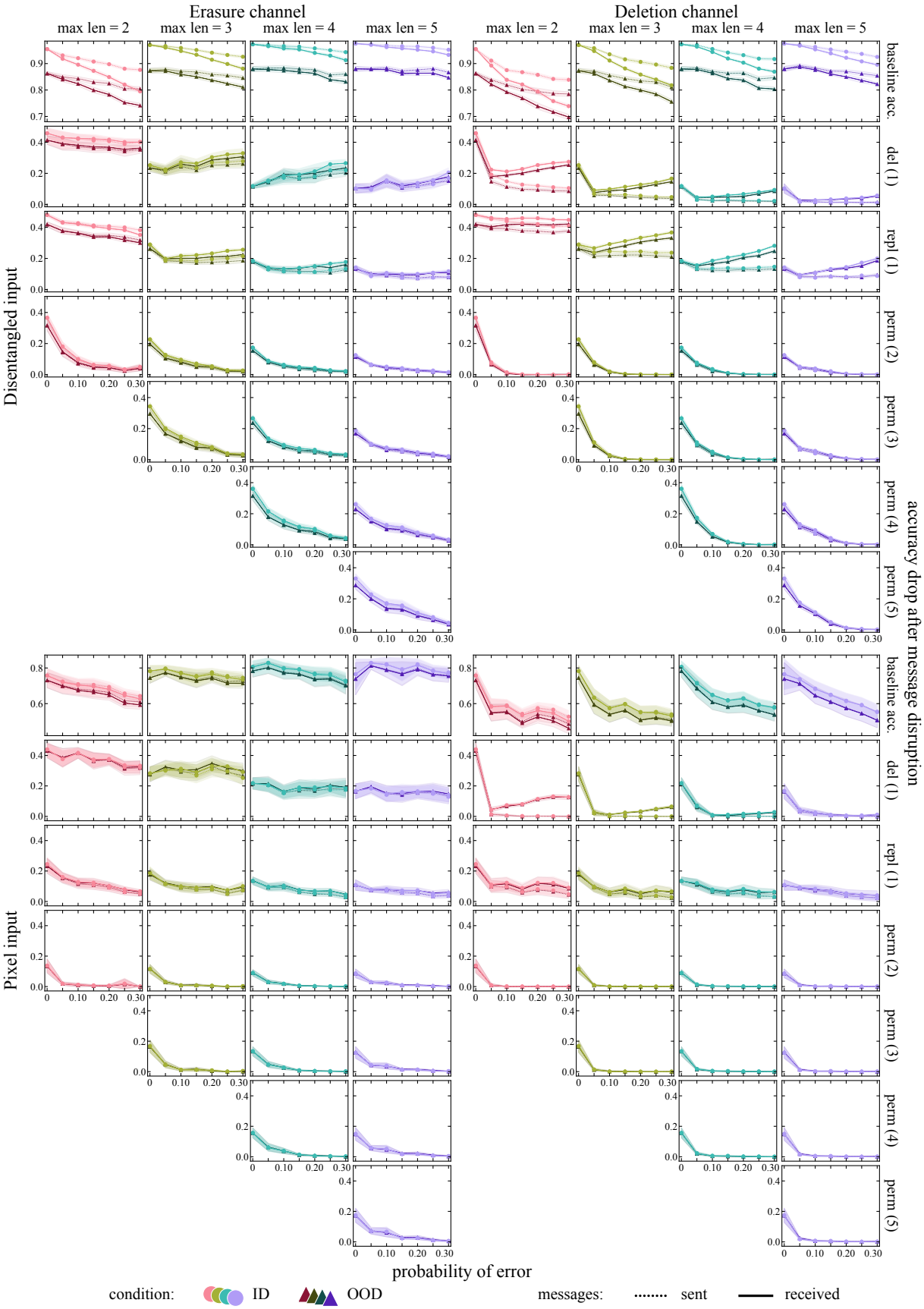
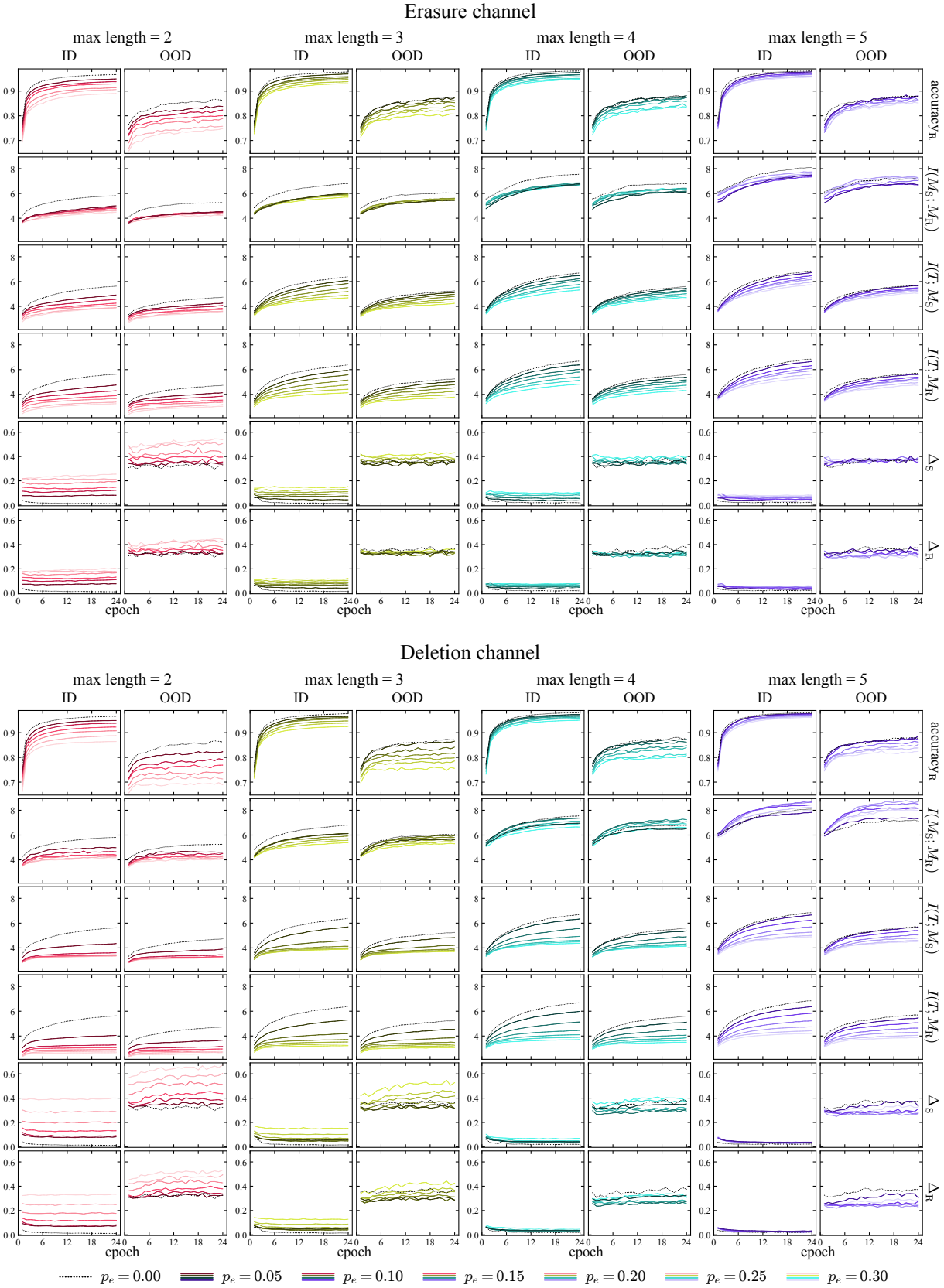
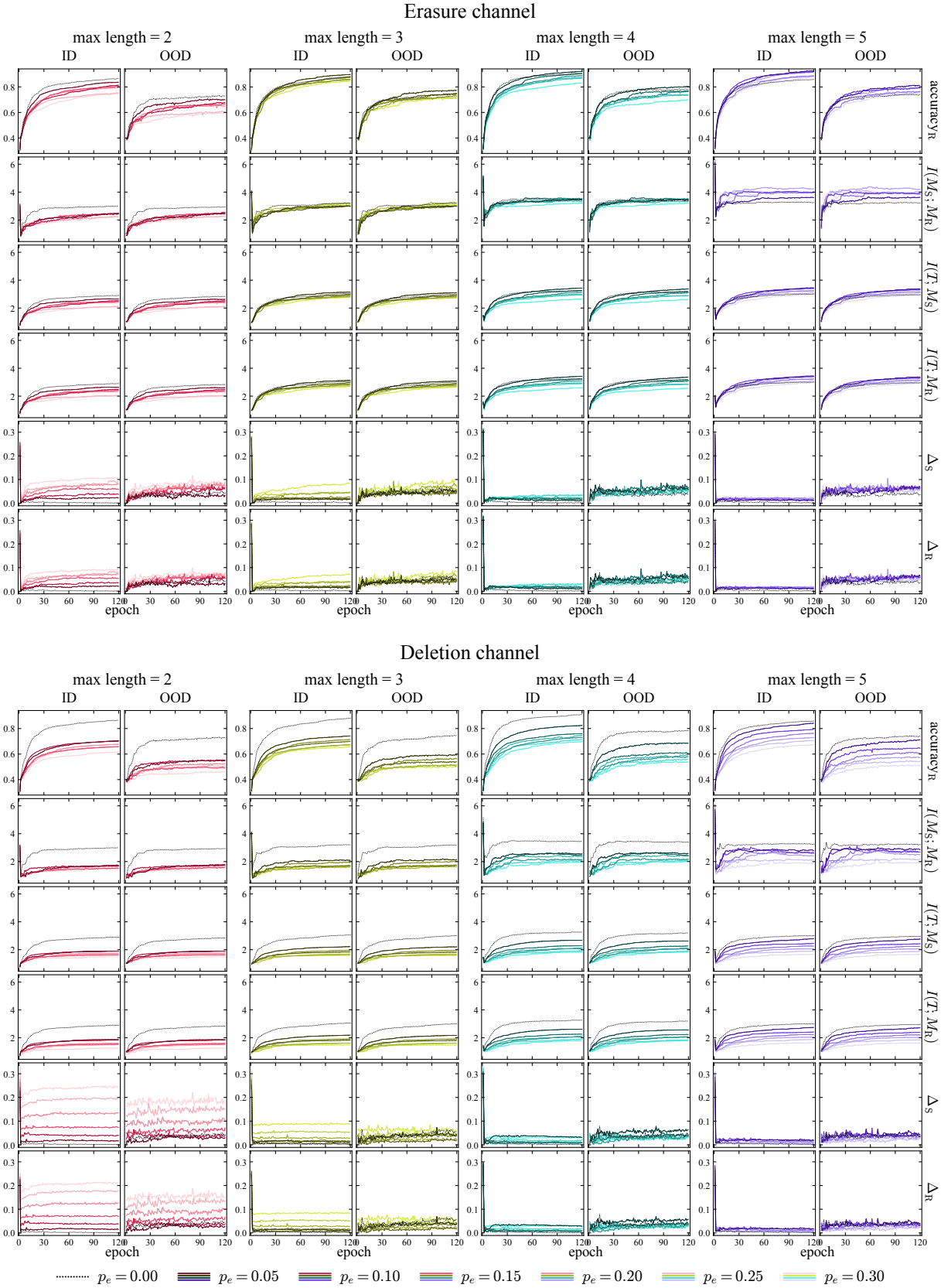


Figure 12: Accuracy after message deletion, replacement or permutation of symbols. The number in parenthesis represents the number of affected symbols. See section E for details.



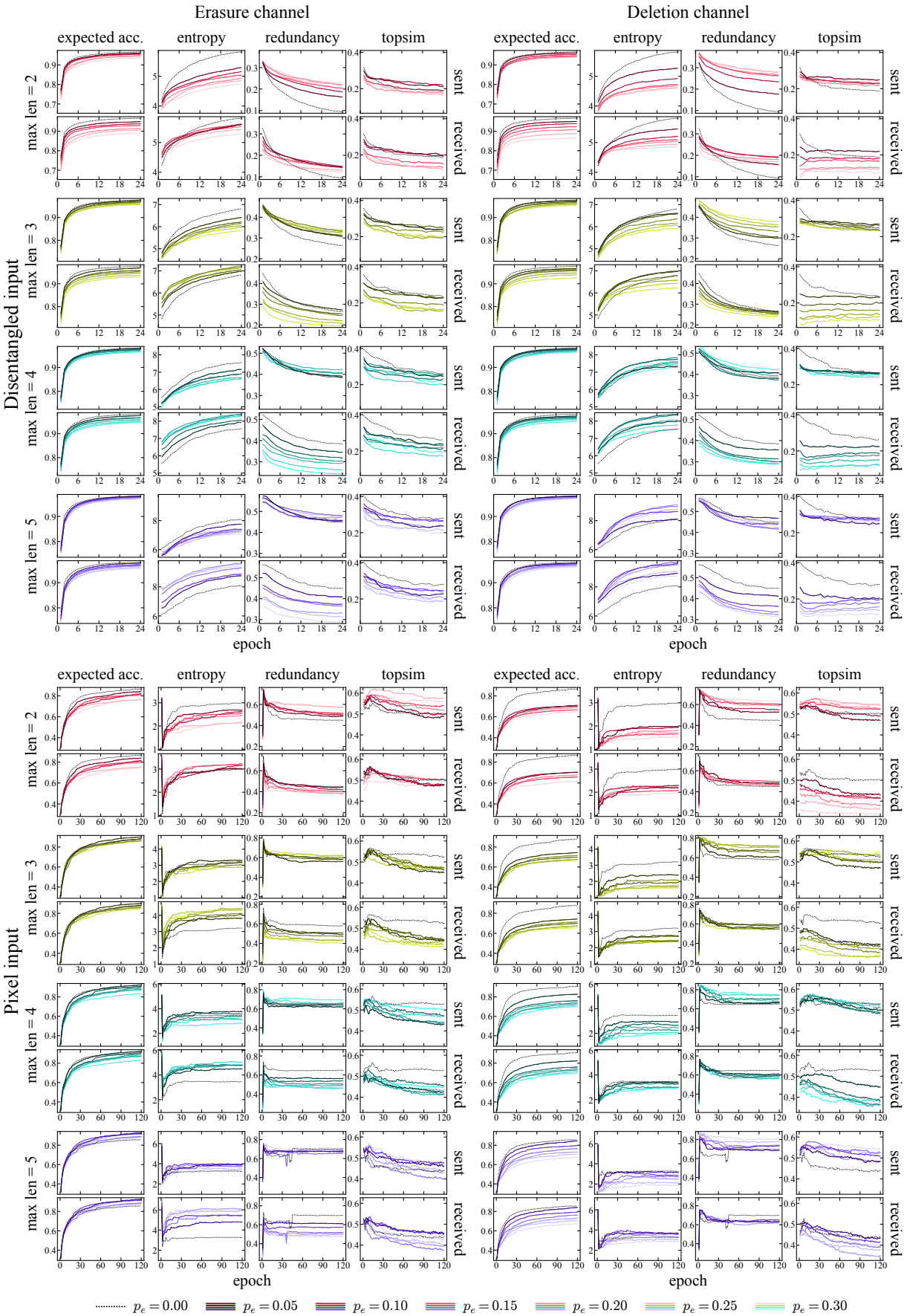
(a) Disentangled input

Figure 13: Changes of information-theoretic measures during training. On the train set, we report expected accuracy computed from the relaxed symbol distributions. $\Delta_S = I(T; M_S) - I(T'; M_S)$, and Δ_R is defined analogously.



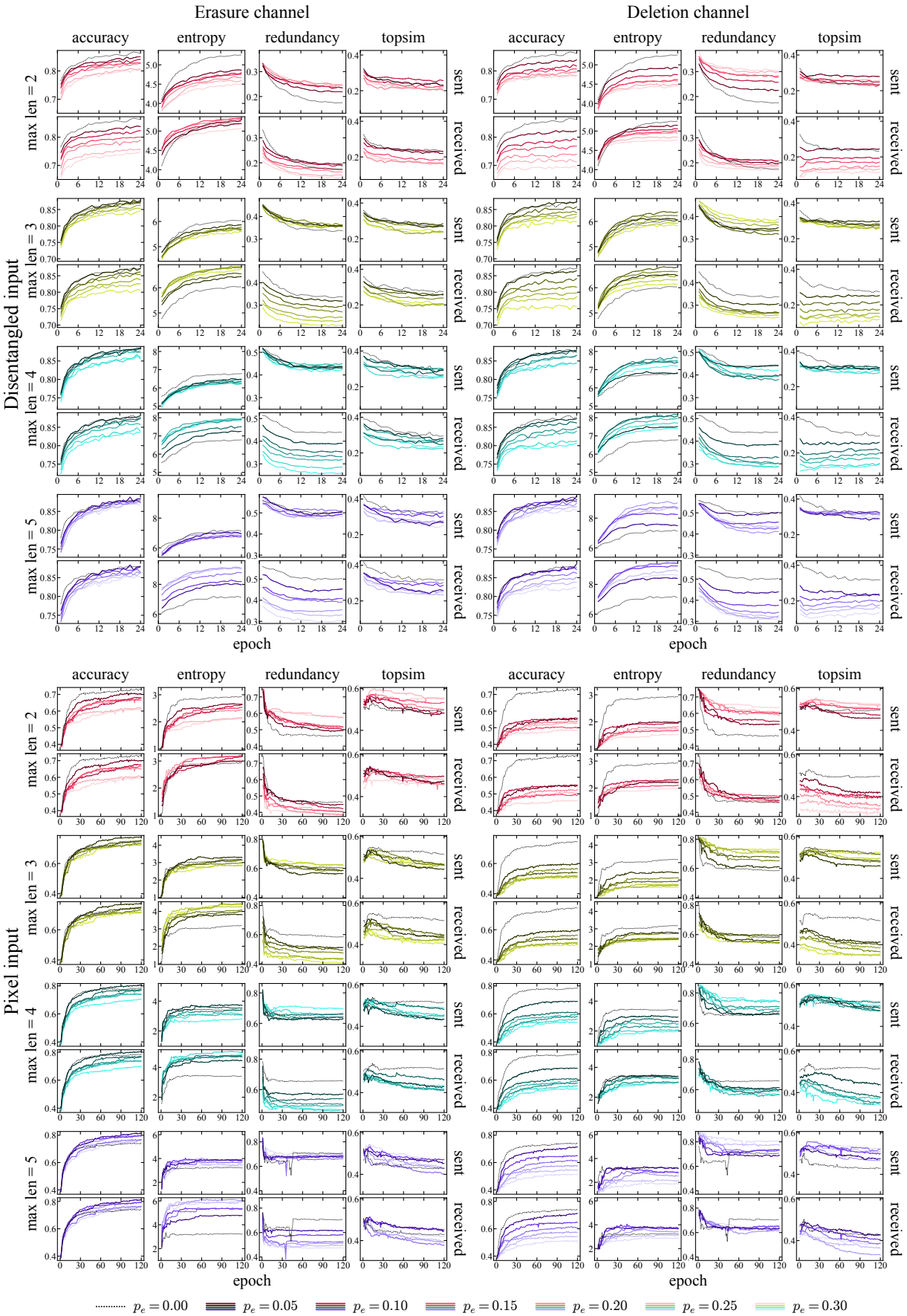
(b) Pixel input

Figure 13: Changes of information-theoretic measures during training. On the train set, we report expected accuracy computed from the relaxed symbol distributions. $\Delta_S = I(T; M_S) - I(T'; M_S)$, and Δ_R is defined analogously.



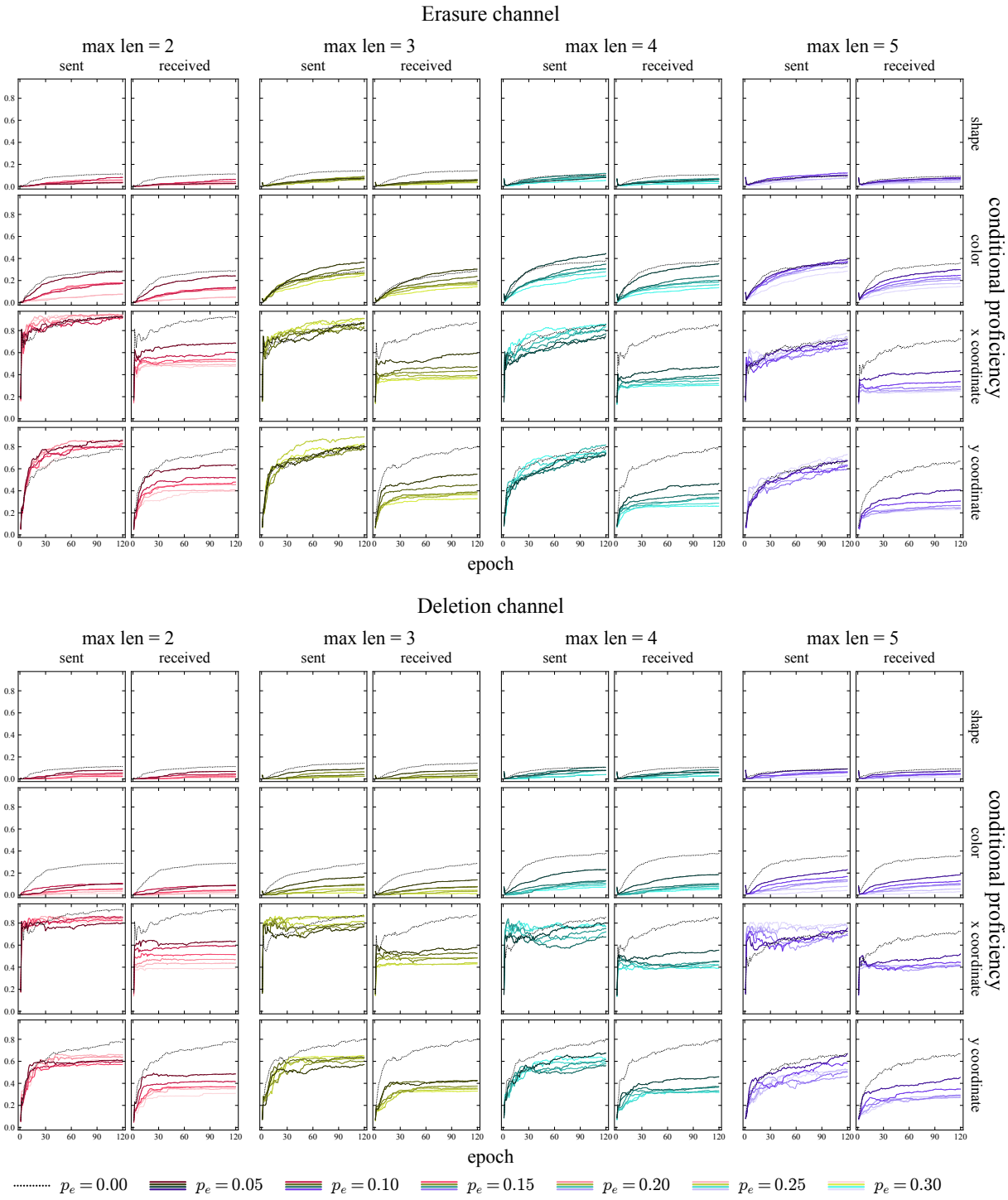
(a) In-domain evaluation

Figure 14: Accuracy, entropy, redundancy and topographic similarity of sent and received messages across training.



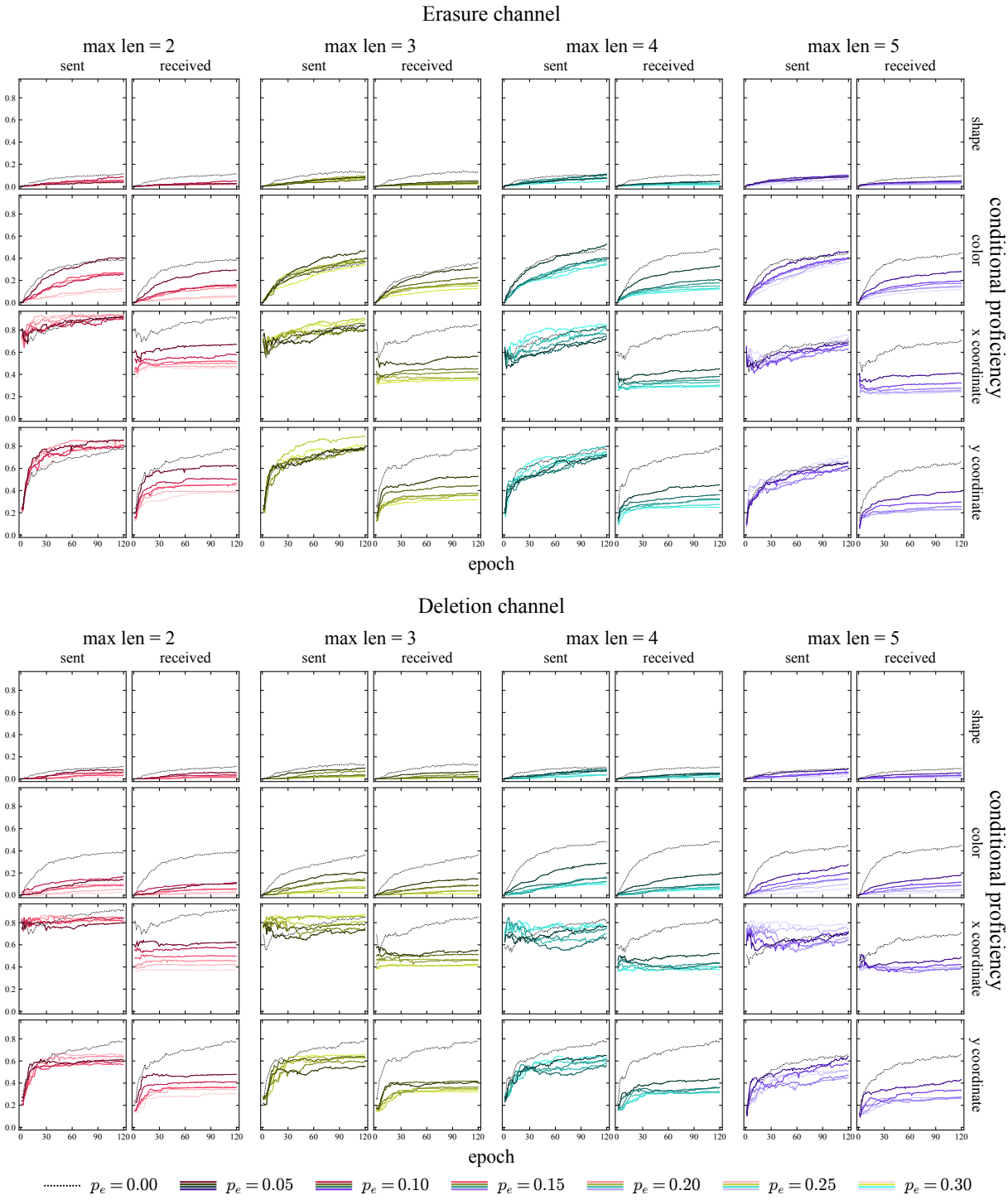
(b) Out-of-domain evaluation

Figure 14: Accuracy, entropy, redundancy and topographic similarity of sent and received messages across training.



(a) In-domain evaluation (pixel input)

Figure 15: Proficiency of each attribute given messages, conditioned on the remaining attributes.



(b) Out-of-domain evaluation (pixel input)

Figure 15: Proficiency of each attribute given messages, conditioned on the remaining attributes.

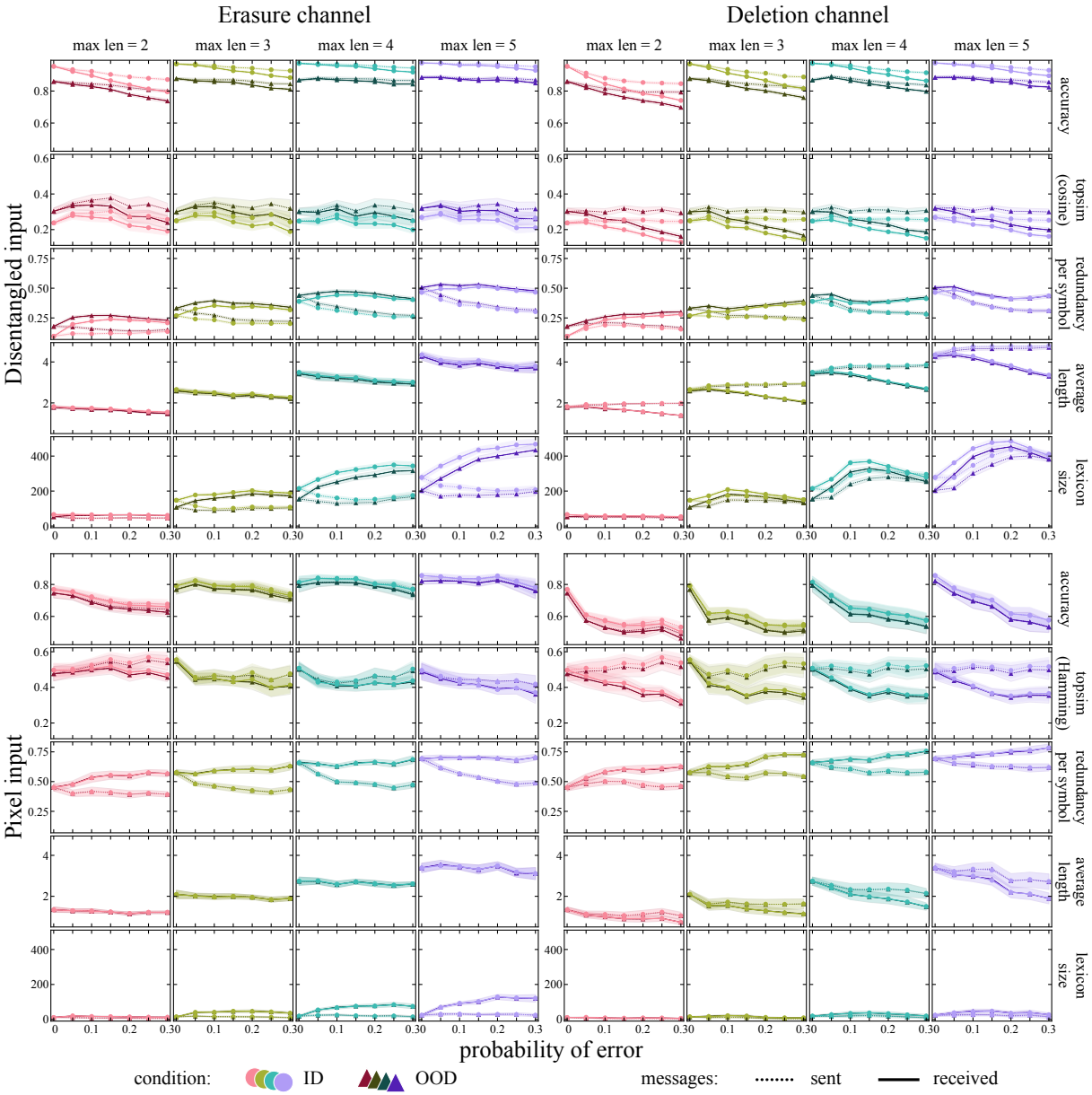


Figure 16: Accuracy, topographic similarity, redundancy per symbol, average message length, and lexicon size, averaged over 20 models trained **with zero length cost**. Shaded areas represent 95% CIs.