The Emergence of Argument Structure in Artificial Languages

Tom Bosc
Mila
Université de Montréal, Canada
bosct@mila.quebec

Pascal Vincent
Meta AI, Mila
Université de Montréal, Canada
CIFAR AI Chair
vincentp@iro.umontreal.ca

Abstract

Computational approaches to the study of language emergence can help us understand how natural languages are shaped by cognitive and sociocultural factors. Previous work focused on tasks where agents *refer* to a single entity. In contrast, we study how agents *predicate*, that is, how they express that some relation holds between several entities. We introduce a setup where agents talk about a variable number of entities that can be partially observed by the listener. In the presence of a least-effort pressure, they tend to discuss only entities that are not observed by the listener. Thus we can obtain artificial phrases that denote a single entity, as well as artificial sentences that denote several entities. In natural languages, if we ignore the verb, phrases are usually concatenated, either in a specific order or by adding case markers to form sentences. Our setup allows us to quantify how much this holds in emergent languages using a metric we call *concatenability*. We also measure *transitivity*, which quantifies the importance of word order. We demonstrate the usefulness of this new setup and metrics for studying factors that influence argument structure. We compare agents having access to input representations structured into pre-segmented objects with properties, versus unstructured representations. Our results indicate that the awareness of object structure yields a more natural sentence organization.

How do languages emerge and evolve? Zipf (1949) viewed language as the result of an optimization procedure balancing information transmission maximization and effort minimization. This view is amenable to formalization and simulation. An early example is Hurford's (1989) comparison of language acquisition strategies, assuming that communication success gives an evolutionary advantage. More generally, subsequent research uses optimization procedures and evolutionary mechanisms to create and study artificial languages (Steels, 1997; Lazaridou and Baroni, 2020).

Such approaches are mainly used with two objectives in mind: Firstly, to improve natural language processing methods; secondly, to help us understand the roles of cognitive and sociocultural factors on the shape of languages, such as our drive to cooperate, pragmatic reasoning, and imitation (Tomasello, 2010).

In the deep learning era, language emergence researchers have focused on the *referential* function of language, namely, how agents communicate about single objects, using artificial noun phrases equivalent to ''blue triangle'' or ''red circle'' (Lazaridou et al., 2017; Kottur et al., 2017). In contrast, we propose to study the *predication* function of language, that is, the expression of relations between entities (*events*). How do artificial agents express events such as ''the blue triangle is above the red circle''?

We introduce an experimental setup for studying predication. The speaker communicates about an event involving a variable number of entities that are in a certain relation. Then, the listener tries to reconstruct this event. To simplify, the relation is observed by both agents.

Crucially, the listener is given a partial observation of the event, ranging from nothing to all but one entity. In the presence of shared context, it is unnecessary for the speaker to communicate the whole event, and a least-effort penalty encourages parsimony. Thus we obtain utterances that refer to single entities in isolation, like phrases, and utterances about several entities, like sentences.

Using these artificial phrases and sentences, we can compute various metrics to quantify compositionality (Szabó, 2020) at the sentence level. A simple sentence typically contains a few phrases that refer to entities. These phrases can generally be understood in isolation, a property sometimes called *context-independence* (Bogin et al., 2018).

Transactions of the Association for Computational Linguistics, vol. 10, pp. 1375–1391, 2022. https://doi.org/10.1162/tacl_a_00524
Action Editor: Daniel Gildea. Submission batch: 3/2022; Revision batch: 6/2022; Published 12/2022.

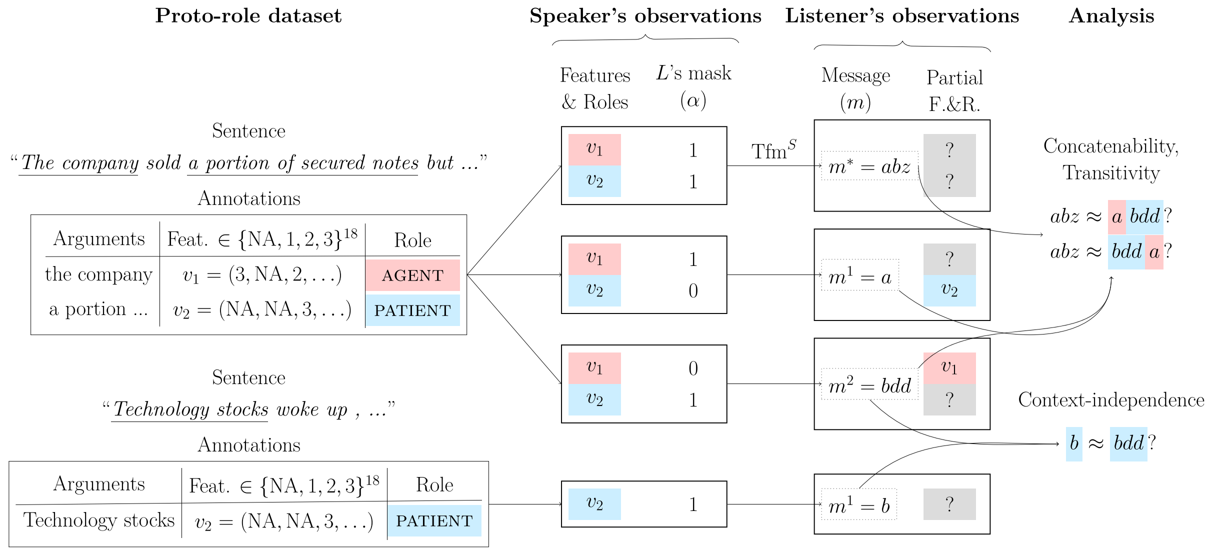

Figure 1: Overview of experimental setup. (From left to right) Proto-role dataset contains annotations (18 features and a role) for each argument and a relation (SELL.01 and WAKE.02, respectively, observed by both speaker S and listener L). **Preprocessing:** From the 1st annotation, 3 datapoints are created, where the number of entities observed by L varies (see *L's mask* and *Partial F.&R.* columns). The 2nd annotation contains a single object so a single datapoint is created. **Training:** S produces a message. L reads it, and the pair of agents S, L is jointly trained to minimize the reconstruction error and the length of the message. As a result of the objective, S only talks about the entities not observed by L. **Analysis:** Informally, *concatenability* measures how concatenation of messages $m^{12} = m^1 \oplus m^2$ and/or $m^{21} = m^2 \oplus m^1$ are interchangeable with the actually sent message m^*; *transitivity* measures how much one order is preferred compared to the other across the dataset (cf. Sections 5, 6).

Moreover, the sentence is the concatenation of these phrases along with the verb. Correspondingly, we introduce *concatenability* metrics that should be large for natural languages. Furthermore, we propose *transitivity* metrics to quantify the importance of word order. A high-level overview of our setup is shown in Figure 1.

This setup enables us to analyze artificial languages without segmenting messages into constituents. Segmentation introduces another layer of complexity, to the extent that in practice, it is not done at all: It is implicitly assumed that each symbol is independently meaningful. However, this assumption is flawed, because if letters or phonemes are assumed to bear meaning, no natural language is compositional.

Previous work has highlighted the influence of input representations and architectures for language emergence. Inappropriate representations completely hinder evolution of a non-trivial language with more than 2 words (Lazaridou et al., 2017) or prevents agents from solving the task altogether (Guo et al., 2019). This suggests that specific inductive biases are still lacking for artificial agents to develop languages like ours.

We posit that the perception of objects as wholes with properties is an important inductive bias. To be able to produce sentences containing referential phrases, it seems that agents need to be able to attend to the referents of these phrases reliably, to conceive of them as bounded objects with intrinsic properties in the first place.

We demonstrate the usefulness of our setup and our metrics for testing this hypothesis. We implement an object-centric inductive bias using attention (Bahdanau et al., 2014) over representations of objects. We compare it to an architecture which disregards the structure of the input, considering it merely a large unstructured feature vector. The object-centric architecture yields more natural languages—they are more concatenable. Furthermore, word order matters more with this architecture than for the baseline. These results are corroborated by our quantitative analysis and measures of generalization outside of the training data.

Our contributions are two-fold. Firstly, on the methodological front, we propose and motivate a novel task and two new metrics. This task not only explains the emergence of compositionality from a functional perspective, but also enables us to easily analyze the learned language, avoiding the problem of segmentation.

Secondly, we provide evidence that when representations reflect the perception of objects as wholes with properties, emergent languages are more natural than when they do not. With this finding we hope to foster the use of more cognitively plausible input representations for explaining language emergence.

1 Task

We design a task for studying how artificial agents predicate. It is an instance of a reconstruction task (Lazaridou et al., 2017), where one agent, the *speaker*, observes an input and produces a message—a sequence of symbols. The message is then read by another agent, the *listener*, who tries to reconstruct the input observed by the speaker.

We train several pairs of agents and study the messages produced by the speakers. This training procedure models language evolution and language acquisition at once, unlike frameworks like iterated learning (Kirby and Hurford, 2002).

The main novelty of our task is that agents are trained to communicate about a variable number of entities. In this section, we explain how the inputs of the agents are obtained by preprocessing the proto-role dataset (Reisinger et al., 2015). Then, we argue that our task is realistic, yet simple enough to permit an easy analysis of the messages.

1.1 The Proto-role Dataset

The data that are fed to agents are based on the proto-role dataset built by Reisinger et al. (2015). This dataset was created to evaluate Dowty's (1991) linking theory, a theory that predicts how verb-specific roles are mapped to grammatical relations in English.

To illustrate the annotation scheme, we use the example from Figure 1, ‘‘the company sold a portion of secured notes’’.

Firstly, a relation is extracted. Here, the verb ‘‘sold’’ corresponds to the PropBank (Kingsbury and Palmer, 2002) label SELL.01, which identifies the verb and its particular sense.

There are $n_{obj} = 2$ arguments of the verb, ‘‘the company’’ and ‘‘a portion of secured notes’’. Each of these arguments is annotated with $n_{feat} = 18$ features indicating various properties of the referred entity. For instance, the first feature indicates whether the entity caused the event to happen, the second feature whether the entity chose to be involved in the event, and so forth (Reisinger et al., 2015). In this work, the meaning of these features is irrelevant. These features are encoded on a Likert scale from 1 and 5 or take a *non-applicable* (*NA*) value. Since the description of each entity is a small feature vector, many different noun phrases correspond to the same feature vector. Thus ‘‘Technology stocks’’ and ‘‘a portion of secured notes’’ in Figure 1 denote the same entity.

Moreover, each argument is also assigned one of six mutually exclusive classical θ-roles. In the example, the arguments respectively have the θ-roles AGENT and PATIENT.

We define an *event* as (i) a relation and (ii) a set of pairs of feature vectors and θ-roles.

1.2 Task Description

For each event in the proto-role dataset, we gather the relation, and for each entity, their 18 features and their role. The features are rescaled from $\{1, 2, 3, 4, 5\}$ to $\{1, 2, 3\}$, and we only retain the arguments in the 3 most frequent θ-roles (AGENT, PATIENT, and a MISC category containing instruments, benefactives, attributes).

The speaker observes the following quantities:

- the PropBank relation β,
- entity features $I^S \in \{\text{NA}, 1, 2, 3\}^{n_{obj} \times n_{feat}}$,
- entity roles $r^S \in \mathcal{R}^{n_{obj}} = \{\text{AGENT, PATIENT, MISC}\}^{n_{obj}}$,
- the listener's mask: $\alpha \in \{0, 1\}^{n_{obj}}$.

The tensors I^S, r^S, and α are indexed by an integer between 1 and n_{obj}, so they represent a set E^S of n_{obj} triplets where each triplet (I^S_i, r^S_i, α_i) characterizes the i-th entity.

The i-th entity is said to be *hidden* iff $\alpha_i = 1$. Hidden entities are not observed by the listener, and the mask α indicates this to the speaker. Since the listener tries to reconstruct the inputs of the speaker, the mask essentially flags the entities that the speaker should communicate about. Thus, the listener observes:

- the PropBank relation β,
- partial entity features $I^L[1 - \alpha]$,

- partial entity roles $r^L[1-\alpha]$,
- the speaker's message: $m \in \mathcal{M}$.

Here, $u[v]$ denotes the tensor obtained by restricting u to the rows i such that $v_i = 1$.

The message space $\mathcal{M}$ is defined as follows. Let $\mathcal{V} = \{1, \ldots, n_V, \text{eos}\}$ be the vocabulary containing n_V symbols, plus an *end-of-sentence* (eos) token. Let n_L be the maximum message length (here, set to $n_L = 8$). $\mathcal{M}$ contains all the sequences of elements of $\mathcal{V}$ with at most length n_L and ending with eos.

A datapoint is valid if $\sum_{i=1}^{n_{obj}} \alpha_i \geq 1$, that is, at least one object is hidden and some information needs to be conveyed. From each event, we add as many valid datapoints as possible to our dataset. In our example, as there are 2 entities, either one or both can be hidden, yielding 3 datapoints.

Given its inputs and the sender's message, the listener tries to reconstruct the sender's inputs. The agents are jointly trained to minimize a reconstruction loss while minimizing the number of symbols exchanged, as formalized in Section 2.2.

1.3 Motivations

All the aspects of the task can have a major influence on the learned languages. In this section, we argue that our task is realistic in important aspects.

Our task is to convey semantic annotations of sentences, not words or sentences directly, because using linguistic data as input could be a methodological mistake. Indeed, language-specific typological properties might leak into the artificial languages.[1] We follow this principle, except for our use of θ-roles. They are linguistic abstractions over relation-specific (participant) roles. This limitation is discussed in Section 8.2.

In our task, agents have to communicate about a realistic, variable number of entities. We posit that this is a crucial characteristic for argument structure to be natural. Indeed, if humans only ever talked about two entities at once, grammar would be simpler since a transitive construction could be used everywhere. In our dataset, the distribution of the number of entities talked about is directly derived from an English corpus, and, to our knowledge, the distribution of the number of arguments does not vary much across languages. Thus we do not expect a bias towards English typology. In Mordatch and Abbeel's (2018) and Bogin et al.'s (2018) works, agents also need to predicate. However, the event structure is unrealistic as it is identical across datapoints: The number of arguments is constant and each argument has the same ''type'' (a landmark, an agent, a color, etc.).

The relation β is observed by both agents. As a consequence, we do not expect artificial sentences to contain the equivalent of a verb. The main reason is that it greatly simplifies the analysis of the artificial languages.

We define the *context* as everything that is observed by both agents: the relation and the non-hidden entities. We now justify why agents share context, and why the loss function includes a penalty to minimize the number of symbols sent (cf. Section 2.2).

First, let us argue that this is realistic. The context is a coarse approximation of the notion of common ground. According to Clark (1996), common ground encompasses the cultural background of the interlocutors, their sensory perceptions, and their conversational history. In theory, the speaker only needs to communicate the information that is not part of the common ground, but transferring more information than needed is not ruled out. However, in practice, humans try to be concise (cf. Grice's [1975] maxim of quantity). The penalty that we use encourages parsimony. It could be seen as the result from a more general principle governing cooperative social activities (Grice, 1975) or even the whole of human behavior (Zipf, 1949).

To illustrate, consider the following situation. Upon seeing a broken window, one would ask ''who/what broke the window?''. A knowledgeable interlocutor would answer ''John'' or ''John did''. In our setup, the speaker is this knowledgeable person, answering such questions about unobserved entities. The context contains the broken window, and the speaker does not need to refer to it since (i) the listener observes it, and since (ii) the speaker knows that the listener observes it (via the mask α). While the speaker *could* still refer to the window, the least-effort penalty makes it costly to do so, so the speaker

[1] For example, if the task was to transmit basic color *terms* instead of, say, color represented as RGB triplets, the choice of a language with only 3 basic color terms vs 11 color terms (as in English) would yield different artificial languages. For one thing, transmitting English color terms would require agents to use more symbols.

avoids it. Even if the agents do not engage in dialogues but in one-time interactions, the mask α can be interpreted as simulating an inference made by the speaker about the listener's knowledge.

This setup is not only realistic, it is also especially useful for the purpose of analysing the emergent languages. By masking all but one entity, we obtain an artificial *phrase* that denotes a single entity. By masking all but two entities, we obtain an artificial *sentence* relating two entities. The metrics that we introduce rely on our abilities to obtain such phrases and sentences. The concatenability metrics can be seen as measures of systematicity, namely, how the meaning of phrases is related to meaning of sentences (Szabó, 2020).

Without this setup, one would need to somehow segment sentences into phrases. To our knowledge, the problem has not been addressed in the language emergence literature, but is identified by Baroni (2020). For instance, applied to English corpora, metrics for quantifying compositionality like Chaabouni et al.'s (2020) disentanglement metrics would tell us that English is not compositional, since single letters are not meaningful.

2 Model and Objective

We present two Transformer-based (Vaswani et al., 2017) variants of the model of the agents: One that encodes an object-centric bias and one that does not. Before delving into their differences, let us describe their common features.

2.1 General Architecture

Both the speaker S and the listener L are implemented as Transformers, each of them built out of an encoder Tfm_e and a decoder Tfm_d.

The inputs of the speaker are encoded into a real-valued matrix V^S, which differs in the two variants of the model. For now, assume that V^S encodes the speaker's inputs and similarly, that V^L encodes the listener's inputs.

The speaker produces a message m by first encoding its input into

$$H = \mathrm{Tfm}_e^S(V^S), \tag{1}$$

then auto-regressively decodes the message

$$m_{t+1} \sim q(m_{t+1}|m_{1:t}, I^S, \alpha, \beta) = \mathrm{Tfm}_d^S(M_{1:t}, H)_t$$

with $M_{1:t}$ the sum of positional and value embeddings of the previously decoded symbols $m_{1:t}$.

At train time, the symbol is randomly sampled according to q, whereas at test time, the most likely symbol is picked greedily. If the maximum length n_L is reached, eos is appended to the message and generation stops. Else, the generation process stops when eos is produced. In order to backpropagate through the discrete sampling, we use the Straight-Through Gumbel estimator (Jang et al., 2016; Maddison et al., 2016).

L also embeds the message m into a matrix M', and its decoder produces a matrix O^L:

$$\begin{aligned} H' &= \mathrm{Tfm}_e^L(M'), \\ O^L &= \mathrm{Tfm}_d^L(V^L, H'). \end{aligned} \tag{2}$$

O^L is then used to predict the presence of the objects as well as all the features of the objects. This computation is slightly different depending on the variant of the models and is detailed below.

Note that Tfm_d^S is invariant with respect to the order of the objects in V^S, since we do not use positional embeddings to create V^S, but rather use the role information directly, as will be explained for each model separately.[2] On the other hand, the message m is embedded using both token and positional embeddings in M and M', so Tfm_d^S and Tfm_e^L are sensitive to word order.

2.2 Loss

The loss function is a sum of two terms, a reconstruction loss and a length penalty.

Reconstruction Loss: The input to reconstruct is a *set*, the set of pairs of 18 features and a θ-roles. For each θ-role, we predict the corresponding features as well as whether an object i in this role is present or not, denoted by γ_i.

For a given data point indexed by j, the reconstruction loss is the sum over all objects i

$$l_j = \sum_i -[\log p(I_i^S|I^L, m, \beta) + \log p(\gamma_i|I^L, m, \beta)].$$

[2] When used without positional embeddings, the encoder of the Transformer is permutation-equivariant, i.e., for any permutation matrix P, $\mathrm{Tfm}_e(PX) = P\mathrm{Tfm}_e(X)$; similarly, the decoder is permutation-invariant in its second argument (the encoded matrix H), i.e., $\mathrm{Tfm}_d(PX) = \mathrm{Tfm}_d(X)$. Permutations are applied to the input matrices, the masks, and the role vectors.

Length Penalty: As done by Chaabouni et al. (2019), we penalize long messages. This can be seen as an implementation of Zipf's (1949) least-effort principle. In its simplest form, the penalty is a term $p_j = \lambda|m_j|$ where λ is a hyperparameter, and $|m_j|$ is the number of symbols in the message.

However, we noticed that messages collapse to empty messages early on during training. This is similar to the well-known *posterior collapse*, where the approximate posteriors of latents of sequence-to-sequence VAEs collapse to their priors (Bowman et al., 2016). We fix the issue by adapting two well-known tricks: Pelsmaeker and Aziz's (2019) minimum desired rate and Kingma et al.'s (2016) free bits. The penalty term becomes

$$p_j = \mathbf{1}_{l_j<\tau}\mathbf{1}_{|m_j|>n_{min}}(\lambda|m_j|),$$

where $\mathbf{1}$ is the indicator function.

For this term to be non-zero, two conditions need to be fulfilled. Firstly, the reconstruction error must be below τ, which is analogous to a minimum desired rate. This threshold can be set without difficulty to a fraction of the reconstruction error incurred by the listener seeing empty messages. In our case, this average error is 18.6. We randomly choose the threshold in $\{5, +\infty\}$ across runs, where $+\infty$ essentially disables the trick.

Secondly, the penalty is above 0 only if the message contains more than n_{min} symbols. This gives models n_{min} ‘‘free’’ symbols for each datapoint. Without this factor, we found that speakers often utter empty messages (in particular, when a single entity is hidden).

For a given data point indexed by j, the total loss to minimize is the sum $l_j + p_j$. During training, the average is taken over a mini-batch ($n = 128$), while during evaluation, it is taken over the entire test split.

2.3 On the Perception of Objects

We demonstrate our setup and metrics by comparing a model which is *object-centric* (OC), that is, aware of objects as wholes with properties, to a baseline model (*flat attention*, or FA), which ignores the structure of the inputs.

We follow Gentner (1982), who argued that perception of objects must be a strong, prelinguistic cognitive bias. She gives the example of a bottle floating into a cave. She imagines an imaginary language in which the bottle and the mouth of the cave are construed as a single entity, and argues that this language would be very implausible. Across languages, the two entities seem to always be referred to by separate phrases, hinting at universals in the perception of objects.

More evidence is provided by Xu and Carey (1996). They showed that infants use spatio-temporal cues to *individuate* objects, that is, to ‘‘establish the boundaries of objects’’. Only around the start of language acquisition do children start to rely on the properties or kinds of objects to individuate. But could it be exposure to language that drives infant to perceive the properties and kinds of objects? Mendes et al.'s (2008) experiments on apes suggest it is the other way around, namely, that linguistic input is not necessary to learn to individuate based on property differences. Thus our hypothesis is that the perception of objects as wholes is a prerequisite for natural language to develop.

To implement the OC bias and the FA baseline, we process the inputs in two ways and obtain different V^L and V^S to plug in Equations 1 and 2. Embedding the matrices I^L and I^S gives us real-valued 3-dimensional tensors. But since Transformers consume matrices, we need to reduce the dimensionality of I^S and I^L by one dimension. It is this dimensionality-reduction step that encodes the inductive biases of OC and FA. We tried to minimize the differences between the two models. Figure 2 shows an overview of their differences.

2.3.1 Object-centric Variant

Let I be either I^S or I^L, where each row I_i represents an object. Each I_i is embedded using a learned embedding matrix Val_j for each feature j, and the result is concatenated, yielding

$$E_i = [\text{Val}_1(I_{i,1})^T; \ldots; \text{Val}_{n_{feat}}(I_{i,n_{feat}})^T].$$

Then, this vector is transformed using a linear function, followed by a ReLU (Nair and Hinton, 2010) and layer normalization (Ba et al., 2016). We obtain $V^{(0)}$, a real-valued $n_{obj} \times d$ matrix with

$$V_i^{(0)} = \text{LN}(\max(WE_i + b, 0)). \qquad (3)$$

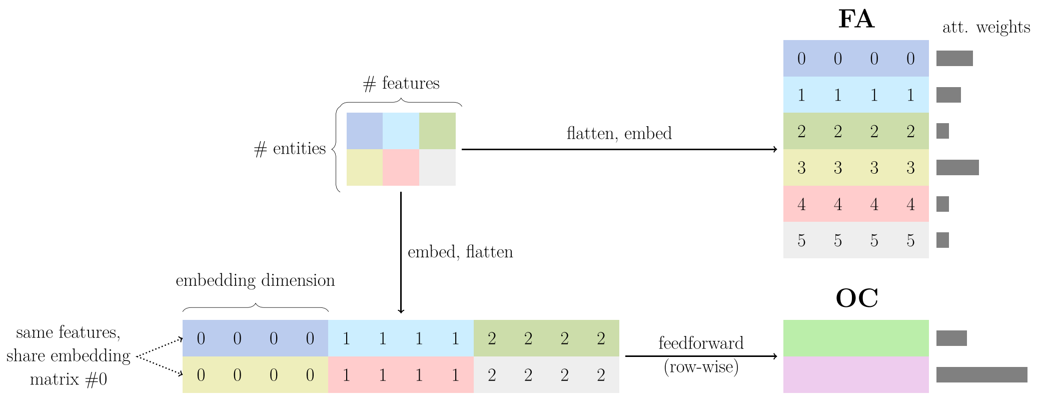

Figure 2: Comparison of flat-attention (FA) and object-centric (OC) variants. The discrete-valued matrices I^L and I^S (upper-left) encode the features of entities. FA turns each datapoint into $(n_{obj} \cdot n_{feat}) \times d$ continuous-valued matrix (with $n_{obj} \cdot n_{feat}$ attention weights), while OC produces a $n_{obj} \times d$ continuous-valued matrix (with n_{obj} attention weights). Numbers index embedding matrices and show weight-sharing. The role information is encoded afterwards and similarly for masking (not shown here).

As for hidden objects and padding objects, they are represented using a single embedding $V_i^{(0)} = v_h$ directly. A *role embedding* is added to this representation to obtain

$$V_i^{(1)} = V_i^{(0)} + \text{Role}(r_i^S).$$

Finally, V is a $(n_{obj} + 1) \times d$ matrix, where d is the size of embedding vectors. V is $V^{(1)}$ with an additional row vector, the β relation embedding.

The listener cannot distinguish between hidden and padding objects, so the message should encode the roles along with the entities' features.

In order to reconstruct the speaker's inputs, the listener linearly transforms each row vector O_i^L (except the one corresponding to the relation) to produce $p(I_i^S|I^L, m, \beta)$, the joint pmf over the discrete features of object i as well as $p(\gamma_i|I^L, m, \beta)$.

2.3.2 Flat Attention Variant

In FA, the structure of the input—composed of different objects with aligned features—is disregarded. Firstly, the input matrices I^S and I^L, where each row corresponds to a single object, are ''flattened''. Secondly, there is one attention weight per feature and object *pair*, instead of a single weight per object as in the OC variant. Finally, each embedding matrix is specific to a role and feature *pair*, instead of being specific to a feature.

Formally, let k be the index of a *pair* of object indexed by i and feature indexed by j. Using a k-specific embedding matrix, we obtain

$$V_k^{(0)} = \text{Val}_k(I_{i,j}),$$

with $V^{(0)}$ a real-valued $(n_{obj} \cdot n_{feat}) \times d$ matrix. Again, hidden and padding objects are represented by a special vector $V_k^{(0)} = v_h$. An *index embedding* is added, similar to the role embedding:

$$V_k^{(1)} = V_k^{(0)} + \text{Idx}(k).$$

As in the OC variant, we obtain V by adding an embedding of the relation β as a row to $V^{(1)}$.

To reconstruct the speaker's inputs, O^L is linearly transformed and to each output vector corresponds a specific feature of a specific object. To predict γ_i, all the output vectors in O^L corresponding to the i-th object are mean- and average-pooled, concatenated and linearly transformed.

3 General Experimental Setup

In the next sections, we review various properties of natural languages, and introduce metrics to quantify these in artificial languages and compare the effect of using OC versus FA on these metrics.

The training set contains 60% of the data, the validation set 10%, and the test set the rest. We denote the entire data set by D and denote by D_k the subsets of D composed of examples for which

$\sum_i \alpha_i = k$, that is, the examples where k objects are hidden.

All the experiments use the EGG framework (Kharitonov et al., 2021) based on the PyTorch library (Paszke et al., 2019).[3] The neural agents are trained using Adam (Kingma and Ba, 2014).

There is a large number of hyperparameters so we resort to random search (Bergstra and Bengio, 2012).[4] Our hyperparameter search is deliberately broad since we do not know a priori which hyperparameter choices are realistic. We expect to obtain results with high-variance, but a major advantage is that we get more robust conclusions by averaging over unknowns.

We perform linear regressions to predict the value of each metric given a binary variable indicating whether OC is used. When the coefficient for this variable is significantly different from 0 according to a t-test, then OC has a significant effect.[5] Additionally, we consider that the entropy of the messages is a mediator that we control for. For instance, the reconstruction error is indirectly influenced by the vocabulary size and the sampling temperature via the entropy. However, if we observe that OC improves the generalization error, we want to exclude the possibility that this is because OC agents send messages with higher entropy than FA agents, since it should be trivial to also increase the entropy of the FA models by modifying hyperparameters.

We discard models with messages of average length below 1 and above 6. Indeed, when the average length is too small, many messages are empty, and when it is too long, artificial sentences are barely or not longer than artificial phrases. These cases are considered a priori unnatural. This leaves us with 100 out of 136 runs.

	Arch.	1 hidden	2 hidden	3 hidden
iD	FA	6.5 ± 1.6	16 ± 3.6	28 ± 5.4
	OC	6.2 ± 1.9	$14 \pm 3.7^{***}$	$25 \pm 5.6^{**}$
OoD	FA	8.9 ± 2.1	24 ± 3.9	41 ± 5.5
	OC	8.3 ± 2.4	$21 \pm 4.6^{**}$	39 ± 5.9

Table 1: Mean and stdev of test reconstruction loss, in distribution and out of distribution. rows: models; columns: # of hidden entities. OC agents generalize better than FA agents. (*: p-value < 0.05, **: p-value < 0.01).

Note that the length penalty works as expected. Without the penalty, the messages all contain the maximum number of symbols. With the penalty, the average message length grows as the speaker needs to send more and more information (on D_1: 4.19, D_2: 5.24, D_3: 5.89).

4 Generalization Performance

Natural languages are often said to be productive and systematic: There is an infinity of utterances which we can understand without having encountered them before (productivity), in particular when we understand constituents of the novel sentence in other contexts (systematicity) (Szabó, 2020). Do emergent languages exhibit these characteristics? In this section, we study such generalization abilities. We measure how well the listener can reconstruct the inputs when the sender communicates about datapoints unseen at train time.

Firstly, we test our models in distribution. Secondly, we test our models *out of distribution* (OoD), following Lazaridou et al. (2018). We compute the empirical marginal distributions over the number of hidden entities, the entities, the roles, and the relations. Then, the OoD test set is sampled from these marginals *independently*.

We measure the reconstruction losses on subsets where 1, 2, and 3 entities are hidden for a finer-grained analysis.

Results: Table 1 contains the results. As expected, performance degrades when we evaluate out of distribution. More interestingly, OC models perform better than FA models both in distribution and out of distribution.

However, the performance difference between OC and FA does not tell us much: Both OC and FA agents could exchange messages that are structured in very unnatural manners. In the next

[3] The proto-role dataset is available here: http://decomp.io/projects/semantic-proto-roles/. The code (including on-the-fly preprocessing of the dataset) is available at https://github.com/tombosc/EGG_f/tree/r1/egg/zoo/vd_reco.

[4] Hyperparameters (uniformly sampled): # Transformer layers $\in \{1,2,3\}$, and dimensions $\in \{200,400\}$, dropout $\in \{0.1,0.2,0.3\}$, Gumbel-Softmax temperature $\in \{0.9,1.0,1.5\}$, $\lambda \in \{0.1,0.3,1,3,10\}$, $n_{min} \in \{1,2\}$, $\tau \in \{5,+\infty\}$. Adam's parameters: $\beta_1 = 0.9$, $\beta_2 \in \{0.9,0.99,0.999\}$.

[5] We manipulate data using the pandas package (The Pandas Development Team 2021; McKinney, 2010), and perform linear regression with the statsmodel package (Seabold and Perktold, 2010). We use HC3 covariance estimation to deal with heteroskedasticity (MacKinnon and White, 1985; Long and Ervin, 2000).

two sections, we introduce metrics to shed light on how the information about different entities is packaged into a single message.

5 Concatenability

In natural languages, the verb encodes the relation while arguments refer to entities, but roles do not have direct equivalents in all languages. They are encoded using three strategies, typically using a mix of strategies within a single language.

In analytic languages like English or Chinese, roles are encoded in word order and possibly using adpositions or clitics, but the role does not change the form of the arguments. For example, in sentences (1a) and (1b), the arguments are identical but are placed in a reverse order, so that their roles are inverted, too:

(1) a. The old lady walks the dog.

b. The dog walks the old lady.

In more synthetic languages like Russian or Turkish, case markings code for the roles. In Russian, these markings are suffixes on nouns, adjectives, and so forth, as can be seen in (2a) and (2b):

(2) a. бабушка выгуливает собаку.

b. бабушку выгуливает собака.

Finally, in polysynthetic languages (Caucasian languages, Samoan languages, etc.), arguments typically look like those in analytic languages, but the roles are encoded using markers on the verb.[6] Since, in this work, relations are not communicated by agents, there is no artificial equivalent of the verb. Therefore, this strategy cannot emerge and we consider it no further.

Crucially, simple sentences are obtained by concatenating a verb and one or several noun phrases that refer to entities, whether word order matters or word order does not matter and cases are marked.

For a single event, by varying what information is available to the listener through the mask α, we get messages describing two entities in isolation (phrases) as well as messages describing two entities at once (sentences). For example, consider $(I^S, (1,1,0), r^S, \beta)$ drawn from D_2, the subset of the data with two hidden objects. Let g be the function that transforms this speaker's inputs into a message via greedily decoding, and define

$$m^* = g(I^S, (1,1,0), r^S, \beta).$$

We obtain the messages sent when L observes the first or the second object in isolation as

$$m^1 = g(I^S, (1,0,0), r^S, \beta),$$
$$m^2 = g(I^S, (0,1,0), r^S, \beta).$$

We define *concatenated messages* to be $m^{12} = m^1 \oplus m^2$ and $m^{21} = m^2 \oplus m^1$, where $\oplus$ is the concatenation operator. This is shown in Figure 1. We define P_2 as the empirical distribution on the subset of D_2 such that neither m^1 or m^2 are empty messages, implying that $m^{12} \neq m^{21}$.

As argued above, in natural languages, m^{12} or m^{21} (or both, if word order is irrelevant) should convey information at least as well as m^*. Denote by $l(m)$ the reconstruction loss incurred by L if L had received the message m, that is, $l(m) = -\log p(I^S | I^L, m, \beta)$. Then, *concatenability from the listener's point of view* is defined as

$$C^L = \mathbb{E}_{P_2}[l(m^*) - \min(l(m^{12}), l(m^{21}))].$$

When close to 0, on average, one of the two concatenated messages (or both) is as informative as the message actually uttered by the speaker for reconstructing the inputs.

L can correctly reconstruct S's inputs from a concatenated message that S is unlikely to utter. Inversely, a concatenated utterance can be highly likely for S even if L might fail to reconstruct S's input from it. Therefore, there are actually two symmetrical measures of *concatenability*, one from the point of view of S and the other from the point of view of L. A similar proposition was made by Lowe et al. (2019) in the context of interactive games. They have shown the usefulness of distinguishing these two points of view.

The metric is defined similarly on the speaker's side with a slight subtlety. Since sampled

[6] This presentation is extremely simplified, for example, Bakker and Siewierska (2009)'s paper for why and how these three strategies generally coexist within a single language.

messages have a maximum message length of n_L, the probability of a sequence longer than n_L is 0. However, concatenated messages are sometimes longer than n_L. We define q_∞ as the distribution generated by S without the constraint that probable sequences have length below n_L. We denote the conditional log-probability of a message given a certain input by $u(m) = \log q_\infty(m|I^S, \alpha, \beta)$. Then, *concatenability from the speaker's point of view* is defined as

$$C^S = \mathbb{E}_{P_2}[\max(u(m^{12}), u(m^{21})) - u(m^*)].$$

It is close to 0 when, on average, one concatenation of the two messages (or both) has roughly the same probability as the actual message.

To give an intuition, let us go back to our examples. Take the speaker of an hypothetical language, English without verbs. Suppose that this speaker, when exposed to a given input $x^S = (I^S, (1,1,0), r^S, \beta)$, produces a sentence m^* corresponding to (1a), ''the old lady the dog''. By exposing the speaker to the same input, but by changing the mask to $(1,0,0)$, they produce $m^1 =$ ''the lady'', while using the mask $(0,1,0)$, they produce $m^2 =$ ''a golden retriever''. C^S compares the log probability of m^* with that of $m^{12} =$ ''the lady a golden retriever'' and $m^{21} =$ ''a golden retriever the lady'', whichever is more probable. Since English without verbs is rather concatenable, the speaker judges that m^{12} is roughly as likely as m^* given the inputs. Thus, the value inside the expectation of C^S will be high, close to 0.

Now, take an identical speaker, except that they assign a very high probability to $m'^1 =$ ''a shoebox'', while the new m'^{12} and m'^{21} are unlikely conditioned on x^S. Then C^S will be low and negative. Perhaps (i) ''a shoebox'' has different semantics when it is used alone in a sentence, as compared to when it is used with a second referent; or perhaps (ii) ''a shoebox'' is never used with another referent in a sentence, and the speaker would use ''a lady'' instead. In any case, concatenability for this speaker would be low, which corresponds to the intuition that their language is unnatural and unsystematic.[7]

	$C^L \uparrow$	$C^S \uparrow$
FA	-6.1 ± 3.8	-29 ± 13
OC	$-3.2 \pm 2.4^{***}$	-26 ± 15

Table 2: Mean and stdev of concatenability metrics on OC and FA runs. (i) OC improves concatenability. Arrows indicate optimal direction. (p-values: *: < 0.05, **: < 0.01, ***: < 0.001).

The same illustration holds for C^L, and it can be adapted to show why C^S and C^L should also be high for more synthetic languages.

Results: We measure these metrics on the test set. In our experiments, they always take negative values: the concatenated messages are on average worse than the actual messages. Some models yield values close to 0, but this depends on the choice of hyperparameters.

Table 2 shows that OC largely improves over FA in terms of both C^L and C^S. For instance, the reconstruction losses of OC models go up by 3.1 nats on average when the best concatenated messages are used instead of the actually sent messages. In contrast, FA models incur a loss that is higher by 6.1 nats. Thus, languages obtained using the OC architecture are more natural than those emerging from FA in the sense of concatenability.

6 Word Order

6.1 Importance of Word Order

Concatenability metrics do not distinguish between the use of word order or some sort of case marking strategy. Since both strategies are found in natural languages, we claim that for all natural languages, this metric should be high. But we also want to know what particularly strategy is used, in particular when concatenability is high.

First, note that it is difficult to detect the presence of case markings directly. Even for the simplest forms of morphology, we are hindered by the segmentation problem of identifying the root and the affix, as mentioned in Section 1.3.[8]

[7] This example only illustrates the intuition. In reality, it is not straightforward to apply these metrics on natural language, because they require probability distributions for the agents. We could learn models that map back and forth between the semantics and the ground-truth utterances, but the models would add some bias. Moreover, we only have ground-truth utterances for English and any attempts to use machine translation would add some more bias.

[8] It is generally even more complicated for several reasons: a lexeme can have several roots, each morpheme can simultaneously encode several semantic properties, and the modification of the root can be non-concatenative (Stump, 2017).

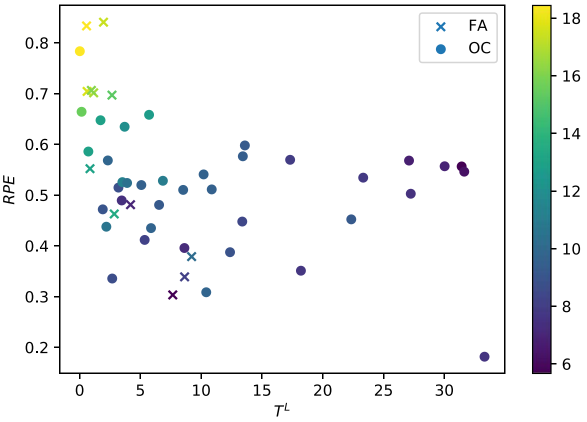

Figure 3: Role prediction error (RPE) as a function of transitivity T^L. Color indicates reconstruction loss. (i) (upper-left quadrant) Low T^L and high RPE implies a high reconstruction error, since roles are not encoded properly. (ii) OC has higher average transitivity than FA, but similar RPE.

	$T^L \uparrow$	$T^S \uparrow$	$RPE \downarrow$
FA	3.4 ± 3.2	10 ± 9	0.48 ± 0.18
OC	$11 \pm 10^{*}$	15 ± 12	$0.52 \pm 0.12^{***}$

Table 3: Mean and stdev of transitivity metrics and RPE for OC and FA. T^L (T^S) statistics and significance computed on runs scoring C^L (C^S) above median. Arrows indicate optimal direction. OC uses word order more than FA. Controls are discussed in the main text. (p-values: *: < 0.05, **: < 0.01, ***: < 0.001).

Yet we can quantify on average how much referential phrases (messages about a single hidden object) encode roles. We train a bigram classifier on the training set and measure its test error, the *Role Prediction Error* (*RPE*). If there are case markings, this error will be low (but the opposite is not true).

Moreover, we introduce two *transitivity* metrics, to directly measure the importance of word order. T^S is defined as:

$$T^S = \mathbb{E}_{P_2}|u(m^{12}) - u(m^{21})|.$$

This metric is 0 if the two concatenated messages are equally probable for S; and it is large if one word order is much more likely than the other for S. Similarly, T^L is defined as

$$T^L = \mathbb{E}_{P_2}|l(m^{12}) - l(m^{21})|$$

and has similar interpretations.

These metrics are only interpretable when concatenability metrics are high enough, so we measured T^S only for runs where C^S is above the median and similarly for T^L.

Results: As can be seen on Figure 3, when transitivity is low and RPE is high, the reconstruction loss is poor (top-left corner), because there is no efficient strategy to encode roles. There is a lot of variance both for OC and FA, but OC models tend to have higher transitivity, both on average and in terms of maximal values. Thus word order is more important for OC runs than for FA runs. This is also confirmed by Table 3.

Table 3 also shows that OC and FA agents have very similar RPE. This means that both encode roles in referential phrases quantitatively similarly. More work is needed to determine how roles are encoded (when they are), that is, if there are traces of morphology or if messages denoting a single entity in different roles are unrelated.

6.2 Consistency of Word Order

To go further, we can study which word orders are favored across different contexts. For every pair of roles such as Agent and Patient, is it the message with the Agent uttered first that is more likely, or the opposite?

To answer the question, instead of looking at the magnitude of the gap as does T^S, we can count which word orders maximize the gap. By finding the most frequent order, we find for each model the *preference* of the speaker P^S, a binary relation on $\mathcal{R}^2$. For example,

$$\{(\textsc{Agent}, \textsc{Patient}), (\textsc{Patient}, \textsc{Misc}), (\textsc{Misc}, \textsc{Agent})\} \quad (4)$$

is such a relation. This is very crude, as it does not distinguish the case where Agent always precedes Patient from the case where Agent precedes Patient 51% of the time, but we leave more involved analyses for future work. We define analogously P^L using the reconstruction loss l instead of message probability u.

Results: We compute preferences P^S and P^L for each run. Out of 100 runs, 29 runs have both C^S and C^L higher than their median values, and 23 of these have equal P^S and P^L.

Entities	α	**A**	**B**	Entities	α	**A**	**B**
$-, 8, 1$	$0, 1, 0$	24, 79, 25	105, 47	$8, 4, -$	1,0,0	79, 24, 24, 79, 24	18, 1, 18
	$0, 0, 1$	105, 16, 105	34, 34		0,1,0	34, 34, 15	15, 34, 15
	$0, 1, 1$	105 , 79, 24	34 , 47		1,1,0	34 , 24, 79, 24, 79, 24	34, 34, 34 , 1, 18
$-, 8, 5$	$0, 1, 0$	24, 79, 25	105, 47	$8, 61, -$	1,0,0	79, 24, 79, 24, 24	18, 18, 19
	$0, 0, 1$	19, 24	18, 18		0,1,0	94, 54, 25, 94, 72	16, 16, 25
	$0, 1, 1$	47, 79, 24, 25	18 , 24		1,1,0	94 , 121, 25 , 79, 24, 79, 24	16 , 19 , 24, 19, 18
$-, 8, 190$	$0, 1, 0$	24, 79, 25	105, 47	$-, 132, 8$	0,1,0	19, 24, 19	19, 59
	$0, 0, 1$	16, 19	19		0,0,1	79, 24, 72	47, 71, 105
	$0, 1, 1$	16 , 79 , 39, 79	105 , 24		0,1,1	24, 19 , 123, 19	18, 24, 59
$-, 8, 39$	$0, 1, 0$	24, 79, 25	105, 47	$-, 287, 8$	0,1,0	35, 19	19, 59, 16
	$0, 0, 1$	16, 44, 16, 72, 2	16, 19		0,0,1	79, 24, 72	47, 71, 105
	$0, 1, 1$	44, 16 , 59, 72	105 , 16		0,1,1	16, 79 , 19, 35	24, 19, 59, 16

Table 4: A sample of messages exchanged about the same entity u_8. Entities: list of entities (''$-$'': no entity; number indicate rank of entity in the dataset; position in the list indicate role: Agent, Patient, Misc). α: mask. **A**, **B**: Messages produced by speakers of models **A** and **B**. Symbols are manually colored to identify phrases (first 2 rows in every block of 3 rows) in artificial sentences (third row in every block). Relations are omitted but are different for each block.

Among all possible relations, some are not transitive, such as (4). However, all the preferences we found are transitive, which is extremely unlikely due to chance. A simple explanation is that transitive relations allows agents to discuss three entities with word order only. However, it does not seem to be universally required by natural languages to have well-defined orders in the presence of many roles. For instance, in English, the use of different prepositions allow for different word order, such as the dative alternation which offers two different orders to talk about three entities.

7 Qualitative Analysis

One can gain intuition about the metrics by looking at messages exchanged by agents. In particular, we compare two models **A** and **B** which both have relatively high concatenability, but **A** has high transitivity scores whereas those of **B** are low. The chosen models also have relatively close reconstruction loss, so that the messages convey roughly as much information about their inputs.

To simplify, we focus on one entity vector and see how it is transmitted when it is in different roles and in different contexts. Since feature vectors are slightly sparse (with many NA values), vectors which have many NAs are sometimes not conveyed at all (the penalty makes it costly to do so). We search for an entity that appears in many different roles and that is sufficiently not sparse. The 8th most frequent vector (u_8) is the most frequent vector that fits these criteria.

First, let us examine the left-hand side of Table 4, which shows how u_8 is talked about in its most frequent role, the Patient role. In both models, u_8 is denoted by the same phrase very consistently (first rows of each block). Thus the context of u_8 (entities and relation) does not seem to influence the message. This property is sometimes called context-independence (Bogin et al., 2018).

Despite using a large vocabulary of 128 symbols, only a few symbols are used. This is due to the difficulty of discrete optimization. We were puzzled to find so many common symbols in the two models, but it turns out that the selected models have the same hyperparameters except for the length-penalty coefficient (**A**: $\lambda = 1$, **B**: $\lambda = 10$).

Each last row of each block of three lines shows an artificial sentence, where two entities are hidden. We can see that most symbols in these sentences also frequently appear in phrases that denote individual entities (identified by their colors). Some symbols from phrases are omitted or in a different order in the sentence, but the precise structure of these phrases is out of scope for our work.

A is more consistent in its use of word order than **B**: **A** almost always refers to Misc before Patient, whereas the order varies for **B**. This is evidence that the transitivity metrics correctly measure the importance of word order, at least when concatenability is high enough.

On the right-hand side of Table 4, u_8 appears in less frequent roles, and we see much more irregularities. Firstly, the phrases denoting u_8 in isolation are less consistent across different contexts (more context-dependence), even though we find a large overlap of symbols. Secondly, we also found more empty phrases (not shown here). Thirdly, we did not find evidence for a lower transitivity of **B** in these roles, but the sample size was smaller.

8 Discussion and Limitations

8.1 Partial Observability and Reference

Thanks to our experimental setup and metrics, we avoid the problem of segmentation. However, concatenability and transitivity rely on a crucial aspect of the task, partial observability, which allows us to obtain messages about a single ''thing'' in isolation. In our case, this ''thing'' is an entity and role pair, but instead, could it be a single attribute like shape or color, as in simpler referential games used in past research?

Such a setup would be similar to our setup (cf. 1.2). However, (i) there would be no relation β; (ii) I^S, I^L and α would be vectors of size n_{feat}; (iii) in terms of models, we would use a simple attention mechanism to select a subset of the features to communicate about.

However, we do not think that this setup realistically models real-life communicative situations. Visual properties like shape and color are often perceived simultaneously. If, sometimes, we fail to perceive colors (for example, at night) or shapes (perhaps due to an occlusion), we rarely need to inquire about these attributes. In general, the missing attributes do not particularly matter, but are useful to identify the *kind* of the entity. For example, the white color and the circular shape of an object tells us that it is a plate, which is useful; but its particular appearance generally does not often matter once it has been categorized. Thus, we generally infer the kind from the observed attributes if possible, or else directly ask for the kind.

By contrast, events are often partially observed, which creates many interrogations. When one observes the consequences of a past action, one often wonders who was the agent that caused it. Similarly, since future events are indeterminate, they are frequently discussed and negotiated. Thus it is frequent to describe events partially.

In sum, the semantics of events are often conveyed partially whereas the semantics of entities are more frequently packaged into the word for a kind. Thus directly transposing this setup to the referential case seems unrealistic. However, perhaps it could be adapted to a discriminative setup (Lazaridou et al., 2017), where the need to convey partial features of objects is clearer.

8.2 On θ-roles

As inputs to our models, θ-roles are much more salient than any of the 18 features associated with entities: Each θ-role is associated with an entire vector added to the keys and values used by the attention mechanisms (cf. Role and Idx in Sections 2.3.1 and 2.3.2). Moreover, there are only three of them and they are mutually exclusive. For these reasons, it is easy to attend over each of them, which explains why many artificial agents rely on θ-roles to structure their messages.

These θ-roles are groups of verb-specific roles (sometimes called participant roles). For example, the LOVER, the EATER, and the BUILDER verb-specific roles are clustered into the verb-general AGENT θ-role, while the LOVEE, the EATEE, and the BUILDEE roles fall under the PATIENT θ-role. Dowty (1991) shows that some θ-roles can be predicted from a small set of features that are mostly related to physical notions of movement and to causality.[9] However, since humans perceive many more features (for example, shapes, colors, textures, etc.), it is not clear why these particular features are preferred to structure the grammars of natural languages.

To answer this question, we might be able to use pretrained unsupervised learning models as feature extractors (Santoro et al., 2017; van Steenkiste et al., 2018; Kipf et al., 2018). An object-centric model like R-NEM (van Steenkiste et al., 2018) can extract object representations from videos of physically interacting objects. An interaction model like NRI (Kipf et al., 2018) can infer the relations between objects given object representations over time, such that these relations are predictive of how the objects change over time. By combining such models, it may be

[9] These features are precisely the features that are used in this paper to represent the semantics of the entities, but their meaning is irrelevant in this work.

possible to learn object, relation, and role representations from videos. We could then use such learned representations as inputs in our communication games to study whether verb-general roles emerge.

9 Conclusion

We have presented an experimental setup for studying how probabilistic artificial agents predicate, that is, how they convey that a relation holds between entities. In our daily lives, events are partially observed and predication is used to share information about what is not observed, often in a parsimonious manner. Our task and loss realistically reflect this function of language.

At the same time, this setup allows us to directly study argument structure while ignoring the internal structure of phrases. Indeed, we can easily obtain artificial phrases, that is, utterances that refer to single entities, as well as artificial sentences, utterances which express the relation holding between different entities. Then, we can study whether and how artificial phrases are systematically composed to form artificial sentences, via our concatenability and transitivity metrics. Thus we completely sidestep the need to segment artificial sentences into phrases, a complicated problem that is unfortunately ignored in previous works.

More precisely, we have argued that all natural languages should have high concatenability, while transitivity is not necessarily high and merely quantifies the importance of word order.

Equipped with this setup and these metrics, we have compared a cognitively plausible architecture that leverages the structure of the inputs into objects with properties (OC) against an implausible baseline that ignores this structure (FA). Object-centric models yield more natural languages in terms of concatenability, while also relying more on word order. Moreover, they generalize better than their implausible counterparts, both in distribution and out of distribution.

These results confirm the importance of the input representations and of the architectures leading to the discretization bottleneck, also reported by Lazaridou et al. (2017) and Guo et al. (2019). In our experiments, discrete low-dimensional inputs were processed by task-specific architectures. However, we believe that one can use high-dimensional representations obtained from pretrained models, as long as these representations are prelinguistic, as object-centric representations seem to be.

Our methods could be extended to investigate other aspects of sentences. For instance, how would agents convey relations? To answer this question, we could use the representations learned via relational unsupervised learning algorithms as inputs. We could study how different relations are discretized into one or several symbols, perhaps the equivalent of verbs and adverbs. We could also analyze how relation-specific roles cluster in abstract roles (like θ-roles) and structure grammar.

Acknowledgments

Tom Bosc was financially supported for this research by the Canada CIFAR AI Chair Program. We also thank the Mila IDT team for the computational infrastructure, as well as anonymous reviewers and the action editors for their helpful feedback.

References

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450v1*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473v7*.

Dik Bakker and Anna Siewierska. 2009. Case and alternative strategies: Word order and agreement marking. *In The Oxford Handbook of Case, edited by Andrej Malchukov and Andrew Spencer*, pages 290–303. 2009. https://doi.org/10.1093/oxfordhb/9780199206476.013.0020

Marco Baroni. 2020. Rat big, cat eaten! Ideas for a useful deep-agent protolanguage. *arXiv preprint arXiv:2003.11922v1*.

James Bergstra and Yoshua Bengio. 2012. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(2).

Ben Bogin, Mor Geva, and Jonathan Berant. 2018. Emergence of communication in an interactive world with consistent speakers. *arXiv preprint arXiv:1809.00549v1*.

Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew Dai, Rafal Jozefowicz, and Samy Bengio. 2016. Generating sentences from a continuous space. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 10–21, Berlin, Germany. Association for Computational Linguistics. https://doi.org/10.18653/v1/K16-1002

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. https://doi.org/10.18653/v1/2020.acl-main.407

Rahma Chaabouni, Eugene Kharitonov, Emmanuel Dupoux, and Marco Baroni. 2019. Anti-efficient encoding in emergent communication. In *Advances in Neural Information Processing Systems*, volume 32, pages 6293–6303. Curran Associates, Inc.

Herbert H. Clark. 1996. *Using Language*. Cambridge University Press.

David Dowty. 1991. Thematic proto-roles and argument selection. *Language*, 67(3):547–619. https://doi.org/10.2307/415037, https://doi.org/10.1353/lan.1991.0021

Dedre Gentner. 1982. Why nouns are learned before verbs: Linguistic relativity versus natural partitioning. Center for the Study of Reading Technical Report; no. 257.

Herbert P. Grice. 1975. Logic and conversation. *Speech Acts*, pages 41–58. Brill. https://doi.org/10.1163/9789004368811_003

Shangmin Guo, Yi Ren, Serhii Havrylov, Stella Frank, Ivan Titov, and Kenny Smith. 2019. *The Emergence of Compositional Languages for Numeric Concepts Through Iterated Learning in Neural Agents*.

James R. Hurford. 1989. Biological evolution of the Saussurean sign as a component of the language acquisition device. *Lingua*, 77(2):187–222. https://doi.org/10.1016/0024-3841(89)90015-6

Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144v5*.

Eugene Kharitonov, Roberto Dessì, Rahma Chaabouni, Diane Bouchacourt, and Marco Baroni. 2021. EGG: A toolkit for research on Emergence of lanGuage in Games. https://github.com/facebookresearch/EGG.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980v9*.

Durk P. Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. 2016. Improved variational inference with inverse autoregressive flow. In *Advances in Neural Information Processing Systems*, pages 4743–4751.

Paul R. Kingsbury and Martha Palmer. 2002. From TreeBank to PropBank. In *LREC*, pages 1989–1993. Citeseer.

Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. 2018. Neural relational inference for interacting systems. In *International Conference on Machine Learning*, pages 2688–2697. PMLR.

Simon Kirby and James R. Hurford. 2002. The emergence of linguistic structure: An overview of the iterated learning model. *Simulating the Evolution of Language*, pages 121–147. https://doi.org/10.1007/978-1-4471-0663-0_6

Satwik Kottur, José Moura, Stefan Lee, and Dhruv Batra. 2017. Natural language does not emerge 'naturally' in multi-agent dialog. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2962–2967, Copenhagen, Denmark. Association for Computational Linguistics. https://doi.org/10.18653/v1/D17-1321

Angeliki Lazaridou and Marco Baroni. 2020. Emergent multi-agent communication in the deep learning era. *arXiv preprint arXiv:2006.02419v1*.

Angeliki Lazaridou, Karl Moritz Hermann, Karl Tuyls, and Stephen Clark. 2018. Emergence of linguistic communication from referential games with symbolic and pixel input. In *6th International Conference on Learning*

Representations, ICLR 2018, Vancouver, BC, Canada, April 30–May 3, 2018, Conference Track Proceedings. OpenReview.net.

Angeliki Lazaridou, Alexander Peysakhovich, and Marco Baroni. 2017. Multi-agent cooperation and the emergence of (natural) language. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24–26, 2017, Conference Track Proceedings*. OpenReview.net.

J. Scott Long and Laurie H. Ervin. 2000. Using heteroscedasticity consistent standard errors in the linear regression model. *The American Statistician*, 54(3):217–224. https://doi.org/10.2307/2685594, https://doi.org/10.1080/00031305.2000.10474549

Ryan Lowe, Jakob N. Foerster, Y.-Lan Boureau, Joelle Pineau, and Yann N. Dauphin. 2019. On the pitfalls of measuring emergent communication. In *Proceedings of the 18th International Conference on Autonomous Agents and Multi-Agent Systems, AAMAS '19, Montreal, QC, Canada, May 13–17, 2019*, pages 693–701. International Foundation for Autonomous Agents and Multiagent Systems.

James G. MacKinnon and Halbert White. 1985. Some heteroskedasticity-consistent covariance matrix estimators with improved finite sample properties. *Journal of Econometrics*, 29(3):305–325. https://doi.org/10.1016/0304-4076(85)90158-7

Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. 2016. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712v3*.

Wes McKinney. 2010. Data structures for statistical computing in Python. In *Proceedings of the 9th Python in Science Conference*, pages 56–61. https://doi.org/10.25080/Majora-92bf1922-00a

Natacha Mendes, Hannes Rakoczy, and Josep Call. 2008. Ape metaphysics: Object individuation without language. *Cognition*, 106(2):730–749. https://doi.org/10.1016/j.cognition.2007.04.007, PubMed: 17537418

Igor Mordatch and Pieter Abbeel. 2018. Emergence of grounded compositional language in multi-agent populations. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2–7, 2018*, pages 1495–1502. AAAI Press.

Vinod Nair and Geoffrey E. Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, pages 807–814.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Tom Pelsmaeker and Wilker Aziz. 2019. Effective estimation of deep generative language models. *arXiv preprint arXiv:1904.08194*. https://doi.org/10.18653/v1/2020.acl-main.646

Drew Reisinger, Rachel Rudinger, Francis Ferraro, Craig Harman, Kyle Rawlins, and Benjamin Van Durme. 2015. Semantic proto-roles. *Transactions of the Association for Computational Linguistics*, 3:475–488. https://doi.org/10.1162/tacl_a_00152

Adam Santoro, David Raposo, David G. Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. 2017. A simple neural network module for relational reasoning. *Advances in Neural Information Processing Systems*, 30.

Skipper Seabold and Josef Perktold. 2010. statsmodels: Econometric and statistical modeling with Python. In *9th Python in Science Conference*. https://doi.org/10.25080/Majora-92bf1922-011

Luc Steels. 1997. The synthetic modeling of language origins. *Evolution of Communication*, 1(1):1–34. https://doi.org/10.1075/eoc.1.1.02ste

Sjoerd van Steenkiste, Michael Chang, Klaus Greff, and Jürgen Schmidhuber. 2018. Relational Neural Expectation Maximization: Unsupervised Discovery of Objects and their Interactions. In *International Conference on Learning Representations*.

Gregory T. Stump. 2017. Inflection. *The Handbook of Morphology*, pages 11–43. https://doi.org/10.1002/9781405166348.ch1

Zoltán Gendler Szabó. 2020, Compositionality, Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*, fall 2020 edition. Metaphysics Research Lab, Stanford University.

The Pandas Development Team. 2021. pandas-dev/pandas: Pandas 1.2.3.

Michael Tomasello. 2010. *Origins of Human Communication*. MIT Press.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Fei Xu and Susan Carey. 1996. Infants' metaphysics: The case of numerical identity. *Cognitive Psychology*, 30(2):111–153. https://doi.org/10.1006/cogp.1996.0005, PubMed: 8635312

George Kingsley Zipf. 1949. *Human behavior and the principle of least effort*. Ravenio Books.