Survey of Spontaneous Emergent Discrete, Compositional and Point-Symmetrical Signals

Pengfei He
Applied Computational & Mathematical Sciences
University of Washington
hepengfe@uw.edu
Supervised and advised by Professor Shane Steinert-Threlkeld
Co-advised by Professor Hannah Hajishirzi

Abstract

Compositionality is considered one of the most important language properties that enables human to create novel sentences using a limited number of semantics. There are prior works on training neural agents to communicate in a signaling game setting. One framework uses continuous latent space to train using back-propagation, and it showed the spontaneous emergence of discrete messages. In this paper, we compare the discreteness and compositionality in two loss settings, Mean Squared Error (MSE) loss and Cross Entropy (CE) loss. And we discover the new message property, point symmetry, under CE setting.

1 Introduction

Compositionality is the property that the meaning of a whole is a function of the meaning of it parts (Thijsse, 1987), and it empowers language to create novel sentences using limited number of meanings (Chomsky, 1957; Montague, 1970). Neural language models have proven to be effective on many natural language processing tasks since their ability to process large amounts of data but it's still far from perfect (Hupkes et al., 2020). The reason for the imperfection is argued that those models either unable to use composition functions (Pinker, 1996) or at least don't use compositions to solve the tasks (Lake and Baroni, 2017). Therefore, researchers are working on decoding compositionality in various ways.

An artificial setting called signaling games is designed to study the factors affecting the emergence of fundamental properties of natural language (Lewis, 1969; Skyrms, 2010). In this setting, two agents communicate with each other in order to accomplish a common task. One agent, Sender, sees some information and then produces and sends a message to the other agent, Receiver. Receiver needs to take action based on the received message. If the action is coherent with the initial information, then it's considered a success. The successful completion of the task will reinforce the message production and the message interpretation corresponding to the initial information. For instance, the process of the referential game is as follows. First, both Sender and Receiver observes a set of objects, and Sender tries to let Receiver to pick a target object. Therefore, Sender needs to produce and send a message to Receiver. Second, Receiver needs to select an object based on the received message, and if the chosen object is same as the target object, the choice of message for Sender and the interpretation for Receiver are both strengthened (Skyrms, 2010; Lazaridou et al., 2017, 2018; Havrylov and Titov, 2017; Chaabouni et al., 2019).

From the perspective of machine learning, signaling games are essentially an autoencoder model (Lan et al., 2020). The message is the intermediate layer; Encoder is layers before the intermediate layer and plays the role of Sender; Decoder are layers after the intermediate layer and plays the role of Receiver. The training is achieved by penalizing picking objects different from the target ones.

Lan et al., 2020 builds a general auto-encoder framework and focused on two design features, discreteness and displacement, together with compositionality. Discreteness means messages correspond to different meanings that are distinct from each other. Displacement is a feature of two contexts with different property values but the same relative extremes for each property. It's expected communication can succeed even in such a situation.

2 Related work

A line of work aims to avoid introducing stochastic nodes into a computation graph in reinforcement learning (Bengio et al., 2013; Schulman et al.,

2015). In addition to it, Lan et al., 2020 uses auto-encoder, where the messages are in continuous latent space, shows discreteness emerges spontaneously under appropriate setting with sampling method assumption. Plus, it generalizes signaling games to auto-encoder, enabling more varieties in game settings and allowing end-to-end training by standard backpropagation. Based on the framework, we investigate the emergent language's discreteness and compositionality under the same set of settings but under CE loss and compare it to the previous work.

3 Methods

3.1 Function Game

Here we introduce a function game, which is a general communication game setting. Besides the introduced Sender and Receiver, there are several basic components. i) a set of contexts denoted by C. ii) a set of actions denoted by A. iii) a group of functions F. Their relationship can be defined as $f : c \rightarrow a$ where $f \in F, c \in C, a \in A$.

The procedure for one play of a Function Game is as follows.

1. Randomly choose a context $c \in C$ and a function $f \in F$
2. Sender sees c and f, and it sends a message to Receiver
3. Receiver see a possibly different context c' and the sent message. Then it needs to choose an action a'.
4. Both agents are rewarded if and only if $a' = f(c')$.

In our experiment, each concept can be defined more concretely as a referential game (Skyrms, 2010; Lazaridou et al., 2017, 2018; Havrylov and Titov, 2017; Chaabouni et al., 2019). 1) $c, c' \in C$ are encoder context and decoder context respectively. They are composed of a set of objects with a pre-specified number of properties. For instance, we represent a context by a $m \times n$ matrix where m is the number of objects, and n is the number of gradable properties. In a regular setting of shared context, Receiver sees the same context but different order, $c' = \text{shuffled}(c)$. We design it this way to force Receiver to identify the target object solely based on messages. In the setting of non-shared context, $c' \neq \text{shuffled}(c)$ where c' has objects with different properties. Moreover, it's better to explain a from the perspective of belief update games. A represents the updated belief state, the different functions in F represent how to update an agent's beliefs in the light of learning a particular piece of information.

Moreover, our experiment is based on one kind of Function Game called Extremity Game which can incentivize and test rich compositionality (Steinert-Threlkeld, 2018, 2020). In addition to the defined context above, each object can be determined by the combination of extreme values of all gradable properties.

Beyond those definitions, we would like to introduce this game through a concrete example. Say we have a small context with only two objects, and each has two gradable properties. For instance, the context consists of two pencils with properties of length and darkness grade. One pencil has length 10 and darkness grade 0.1, and the other has length 1 and darkness grade 0.2. Therefore, Sender and Receiver both see the two pencils represented by property values.

Suppose we have four pencils with gradable properties of length, l, and darkness grade, d. Each pencil are constructed as $p = (l, d)$. We define 8 pencils and each four of them belong to one context and its extreme features are described on each line.

$$p_1 = (9, 0.2), p_1' = (10, 0.4), \text{“longest”}$$

$$p_2 = (8, 0.1), p_2' = (7, 0.3), \text{“least dark”}$$

$$p_3 = (2, 0.9), p_3' = (4, 0.6), \text{“darkest”}$$

$$p_4 = (1, 0.8), p_4' = (3, 0.5), \text{“shortest”}$$

According to the definition of the strict context where $|c| = 2 \times N$, where N is the number of properties. We let encoder context $c_1 = \{p_1, p_2, p_3, p_4\}$ where the number of objects is two times the number of properties. To identify a pencil, we only need to use an extreme of one property. For example, we want to choose a function called the "darkest" pencil. In shared context, Receiver see the same but shuffled context as Sender, $c_1' = \text{shuffled}(c)$. Then we have a function $\max_d c = p_4$ that extracts the object with the darkest pencil from the input context. However, in non-shared setting, we will have a decoder context with different property values. Let's define a $c' = \{p_1', p_2', p_3', p_4'\}$ where p_i' is still the target object with its extreme value of one property. For example, $p_4' = \max_d c'$ in other

words, p'_4 is still darkest in the new and different context even it has different darkness degree from p_4.

The goal to learn a two-dimensional message that Sender can produce, and Receiver can understand and choose an object following the message. Follow the previous example, the task is successful when Receiver chooses p_4 in strict context and p'_4 in non-strict context.

To introduce non-strictness, we add two additional pencils into c.

$$p_5 = (3, 0.5)$$

$$p_6 = (10, 1)$$

We can then define the new non-strict context $c_2 = \{p_1, p_2, p_3, p_4, p_5, p_6\}$. The non-strictness is two-fold. First, there is no one-to-one correspondence between an extreme property value and one object in the same context (as in the Extremity Game from Steinert-Threlkeld, 2018, 2020). In this example, p_5 doesn't have extreme features for identification, and p_6 has two extreme features "darkest" and "longest" with respect to c_2. None of those two pencils can be identified by single extreme feature. Second, subsequently, the requirement $C = 2 \times N$ is no longer required.

3.2 Model

The model Figure 1 is essentially auto-encoder-decoder architecture which enables Sender to encode features into lower-dimensional intermediate layer and Receiver to reveal the hidden feature of the intermediate layer to accomplish the task of target object recovery. We expect the hidden feature to have properties of discreteness and compositionality. The model architecture is as follows. Both encoders and decoders are multi-layer perceptrons composed by two 64 dimensional hidden layers and rectified linear activation(ReLU). A smaller, intermediate layer connects encoders and decoders without activation function, and it represents the message in latent space. At the end of the structure, it outputs Receiver's prediction.

3.2.1 Game parameters

- Context identity. It's whether each sampling function corresponds to one context. Namely, there are two options, shared and non-shared context. Without considering the object order, in shared context, $c = c'$; in non-shared context, $c \neq c'$.

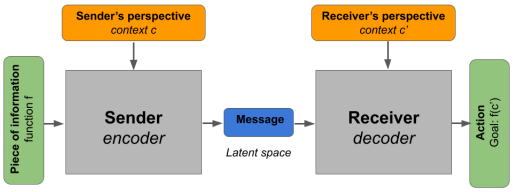

Figure 1: Our model architecture combining concepts of autoencoder and signaling games (Lan et al., 2020).

- Context strictness. In strict contexts, there is a one-to-one correspondence between F and A. In non-strict contexts, an object can be $\arg\min$ or $\arg\max$ of zero or multiple dimensions.

- Loss type. There are two loss type, Mean Squared Error(MSE) and Cross-Entropy(CE). We want to test our experiment soundness under different loss types. Under the setting of MSE, the prediction is object properties representing the properties of the target object. Under the setting of CE, the prediction is a one-hot vector representing the target object in context.

- Message size. It's the dimension of the intermediate layer between two MLP, encoder and decoder. In all experiments, the message dimension is always 2 for visualization.

3.2.2 Training Details

We have trained model for 5000 epochs with the Adam optimizer (Kingma and Ba, 2015) of learning rate 0.01, $\beta_1 = 0.9$ and $\beta_2 = 0.999$. We fed a mini-batch of 128 contexts concatenated with one-hot function selectors. The network loss can be either MSE or CE. For each combination of game parameters, we run 10 trials with different random seeds. The network's loss is either MSE or CE between the target object $f(c')$ and the object generated by Receiver. Specifically, under the CE setting, we randomly sample the predicted object according to the output distribution and then feed it into the loss function. [1]

	MSE		CE	
	Shared	**Non-shared**	**Shared**	**Non-shared**
Strict				
10 objects	$63.33\% \pm 2.08$	$61.89\% \pm 1.41$	$11.86\% \pm 2.49$	$11.76\% \pm 1.04$
Non-strict				
5 objects	$50.20\% \pm 2.30$	$46.43\% \pm 1.97$	$81.87\% \pm 2.91$	$83.00\% \pm 2.52$
10 objects	$33.34\% \pm 1.01$	$32.88\% \pm 1.76$	$30.79\% \pm 6.06$	$27.91\% \pm 4.87$
15 objects	$27.89\% \pm 1.17$	$28.81\% \pm 0.99$	$12.22\% \pm 1.35$	$12.64\% \pm 1.75$

Table 1: Communicative success measured by object recovery accuracy.

	MSE		CE	
	Shared	**Non-shared**	**Shared**	**Non-shared**
Strict				
10 objects	1.00 ± 0.00	1.00 ± 0.00	0.19 ± 0.09	0.23 ± 0.09
Non-strict				
5 objects	0.99 ± 0.03	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00
10 objects	1.00 ± 0.00	1.00 ± 0.00	0.84 ± 0.21	0.75 ± 0.18
15 objects	1.00 ± 0.00	1.00 ± 0.00	0.42 ± 0.20	0.37 ± 0.10

Table 2: Discreteness in production as measured by F1 scores in automatic message clusterization.

	MSE		CE	
	Shared	**Non-shared**	**Shared**	**Non-shared**
Strict				
10 objects	$63.98\% \pm 0.82$	$62.38\% \pm 1.78$	$11.94\% \pm 3.94$	$12.33\% \pm 3.15$
Non-strict				
5 objects	$47.49\% \pm 2.24$	$47.61\% \pm 2.05$	$81.82\% \pm 3.46$	$82.44\% \pm 2.25$
10 objects	$33.03\% \pm 1.80$	$33.34\% \pm 1.17$	$31.69\% \pm 5.85$	$28.94\% \pm 6.29$
15 objects	$28.65\% \pm 1.45$	$29.21\% \pm 1.35$	$12.59\% \pm 1.76$	$14.08\% \pm 1.67$

Table 3: Discreteness in perception as measured by object prediction accuracy using average message from each function cluster.

	Compositionality by Addition		**Composition Network**	
	Shared	**Non-shared**	**Shared**	**Non-shared**
Strict				
10 objects	$12.63\% \pm 4.19$	$8.70\% \pm 3.44$	$6.25\% \pm 2.04$	$7.85\% \pm 3.82$
Non-strict				
5 objects	$15.60\% \pm 2.21$	$16.78\% \pm 2.30$	$17.66\% \pm 3.83$	$13.35\% \pm 2.77$
10 objects	$5.61\% \pm 1.77$	$6.31\% \pm 2.32$	$5.32\% \pm 3.18$	$4.60\% \pm 1.48$
15 objects	$3.52\% \pm 1.51$	$2.47\% \pm 1.26$	$1.47\% \pm 0.96$	$1.91\% \pm 1.65$

Table 4: Training under MSE Loss: communicative success by composition.

	Compositionality by Addition		**Composition Network**	
	Shared	**Non-shared**	**Shared**	**Non-shared**
Strict				
10 objects	$9.10\% \pm 1.02$	$9.91\% \pm 1.49$	$11.30\% \pm 1.35$	$11.50\% \pm 1.26$
Non-strict				
5 objects	$19.02\% \pm 1.34$	$19.37\% \pm 0.76$	$40.15\% \pm 10.28$	$44.58\% \pm 7.81$
10 objects	$9.71\% \pm 1.76$	$8.98\% \pm 1.37$	$20.69\% \pm 3.65$	$18.95\% \pm 3.08$
15 objects	$7.59\% \pm 1.85$	$6.67\% \pm 1.19$	$11.54\% \pm 0.84$	$11.63\% \pm 1.51$

Table 5: Training under CE Loss: communicative success by composition.

4 Results

4.1 Commumicative success

Here we measure communicative success by recovering the target object from decoder contexts. In non-shared context settings, as the context might be different, we consider communicative success as the predicted object is closest to $f'(c)$ compared to all other objects. The results are reported in Table 1. We evaluate accuracy by comparing the success rate with the reference probability, chances of randomly picking, which is $\frac{1}{C}$. Under MSE loss, the model handles displacement fairly well, that the accuracy is well above the reference probability. Under CE loss type, the network's performance on displacement is good as well, considering CE is a weaker loss signal than MSE. However, the model's accuracy degrades when there are more objects to identify. From Table 1, we observe that under the setting of CE loss and the non-strict context of $C = 5$, the communicative success rate is around 82%. However, under the same setting, except for more objects, the rate drops 60% and 70% for $C = 10$ and $C = 15$, respectively. Plus, the rate under CE loss and the strict context of $C = 10$ is unexpectedly low as we expected the strictness could ease the model under the CE setting.

4.2 Discrete Signals

Here we investigate how discrete the messages are under different settings. We measure the discreteness both numerically and visually. Numerically, we analyze the results through Sender's production and Receiver's perception. Plus, we observe the visualization of message clusters as supplementary experimental results to analyze discreteness.

First, we categorize shapes of our clusters descriptively to cover the full phenomenons we observed and label all occurrences in each setting in the following visual analysis. Most phenomenons have correspondences with cluster visualizations in Figure 2. We list the phenomenons in the order of decreasing discreteness.

1. Compact clusters without overlapping.

2. Less compact clusters without overlapping.

3. Flat cloud-like clusters with different degrees of overlapping.

4. Aligned clusters with different degree of overlapping.

5. No clear clusters but messages aligns.

As for Sender's production, we sample 100 contexts and collected the output of Sender for function f. Then we applied unsupervised clustering algorithm to the messages (DBSCAN, Ester et al., 1996, with $\epsilon = 0.5$). The label for each cluster was named by the function most often at the source of a point in the cluster. It enables us to compute F1-score which is reported in Table 2. Under the MSE loss settings, the model achieves a near-optimal F1 clustering score, and cluster shapes are under type 1 and type 2. By comparison, under the settings of CE loss and non-strict contexts, the model has a decreasing F1-score as the number of objects increases, and the cluster shapes are type 1 and type 3. In other words, there is a certain degree of overlapping.

As for Receiver's perception, we study Receiver's perception by feeding unseen messages from message clusters. Then let Receiver identify target objects based on the messages. We sampled ten artificial messages for each cluster during the sampling, take an average, and then get a message that never appeared before to test Receivers' perception. The artificial messages are fed to Receiver for 100 different contexts. The output object accuracy for the messages is given in Table 3. By comparing Table 1 and Table 3, we can conclude that averaged messages from function clusters can represent objects with similar performance compared to the messages themselves even though model under CE settings has relatively lower object prediction accuracy as more objects in the context.

Interestingly, we also found some special phenomenon under the CE settings. First, under the setting of $C = 5$, the clusters under the CE setting (type 1) are more compact than those under the MSE setting (type 2), and it corresponds to about 34% improvement on accuracy prediction using the averaged message in Table 3. Second, under the setting of CE loss and strict and non-shared context, the shapes of clusters are unexpectedly type 3, type 4, and type 5, which all negatively affect model prediction as we expected strict context could ease model training. By checking the loss graph, training loss only decreases around 0.1 for the first 5000 epochs in the setting of a strict context

[1]Code implementation is available at https://github.com/feipenghe/singaling-auto-encoder

of $C = 10$. By comparison, in the non-strict context, the training loss decreases more when there are fewer objects. For the first 5000 epochs, in the setting of non-strict context, training loss decreases approximately 1.4, 0.6, 0.3 for $C = 5$, $C = 10$, $C = 15$ respectively.

Based on the analysis above, two factors affect the prediction accuracy most, whether the context is strict and the number of objects.

4.3 Compositionality

There are two methods to evaluate compositionality. First, traditionally, Mikolov et al., 2013 looks for compositionality on embeddings of the word level, arithmetic properties such that WE(queen)=WE(king)-WE(man)+WE(woman). We would like to test the hypothesis in our game whether message clusters from dimension i and j can have relationship $M(c, \underset{i}{\text{argmax}}) = M(c, \underset{j}{\text{argmax}}) - M(c, \underset{j}{\text{argmin}}) + M(c, \underset{i}{\text{argmin}})$. We calculate the right-hand equation on 100 contexts and feed them to Receiver for each such pair of dimensions. Then we analyze the prediction accuracy of Receiver.

First, as reported in the left two columns of Table 4 and Table 5, MSE and CE settings both suggest there is no such an arithmetic relationship to show compositionality even though model under CE setting shows slightly higher accuracy. Besides the poor performance, this method is also disputable as it has a strong prior considering composition as addition (Linzen, 2016; Chen et al., 2017). Second, we trained an additional composition network composed of 2 hidden layers and 64 ReLU units (Nair and Hinton, 2010; Glorot et al., 2011). The input and output follow the same manner as the first method. It's expected to capture more complex and nonlinear relationships of the three groups of messages without the strong addition prior. As reported in the right two columns of Table 4 and Table 5, the model under the CE setting generally shows better performance than that under MSE loss, which doesn't indicate composition at all by comparing reference probability. Under the setting of the non-strict context of $C = 5$, the prediction accuracy is around 40%, which is around double of reference probability. We think that even though the prediction accuracy under CE decreases as the number of objects increase, the performance is fairly well given the C-based reference probability. The reason is that predicting is more difficult, especially under the CE setting where the loss is only based on whether the output object is the same as the target object rather than comparing property values directly.

Also, in general, models trained using MSE loss drive compositionality away as the communicative accuracy by composition is generally below reference probability. In other words, it's worse than random picking.

Plus, the model under CE loss and strict context still performs pretty badly, which is reasonable as their composition components, message clusters, have low prediction accuracy after training as analyzed in previous sections.

4.4 Point Reflection

There is a pattern that messages trained from min function and messages trained from max function are reflected across the origin, which happens only under CE settings. It gives more structure, which positively correlates with the higher communicative success by composition and is captured by the composition network. By comparing Table 4 and Table 5, under the same settings, model prediction under CE loss generally surpasses model prediction under MSE loss even in cases where the CE model suffers from less discrete clusters and overlapping.

Also, a less ideal observation is that messages from one type of function don't stick to one side of the reflection point. To be more specific, for a property i, function with odd number indices n are max function denoted by $\underset{i}{\text{argmin}}\ F_n$ and function with even number indices are min function denoted by $\underset{i}{\text{argmin}}\ F_{n+1}$. For example, F_0 and F_1 are min and max functions applied on the first property of the same context, and they are reflected by one point close to the origin. The other phenomenon is dimension reduction. Sometimes the clusters are aligned in 2D message space. We think it will adversely affect prediction accuracy as there is less vector space for representation. The reason for the alignment and point reflection remains unknown.

Message developed based on different choice functions differentiate the messages along one dimension under CE loss. Accordingly, when there are more objects, messages don't cluster well according to Table 2. From the picture, we can see there is some overlapping between message data points along that dimension.

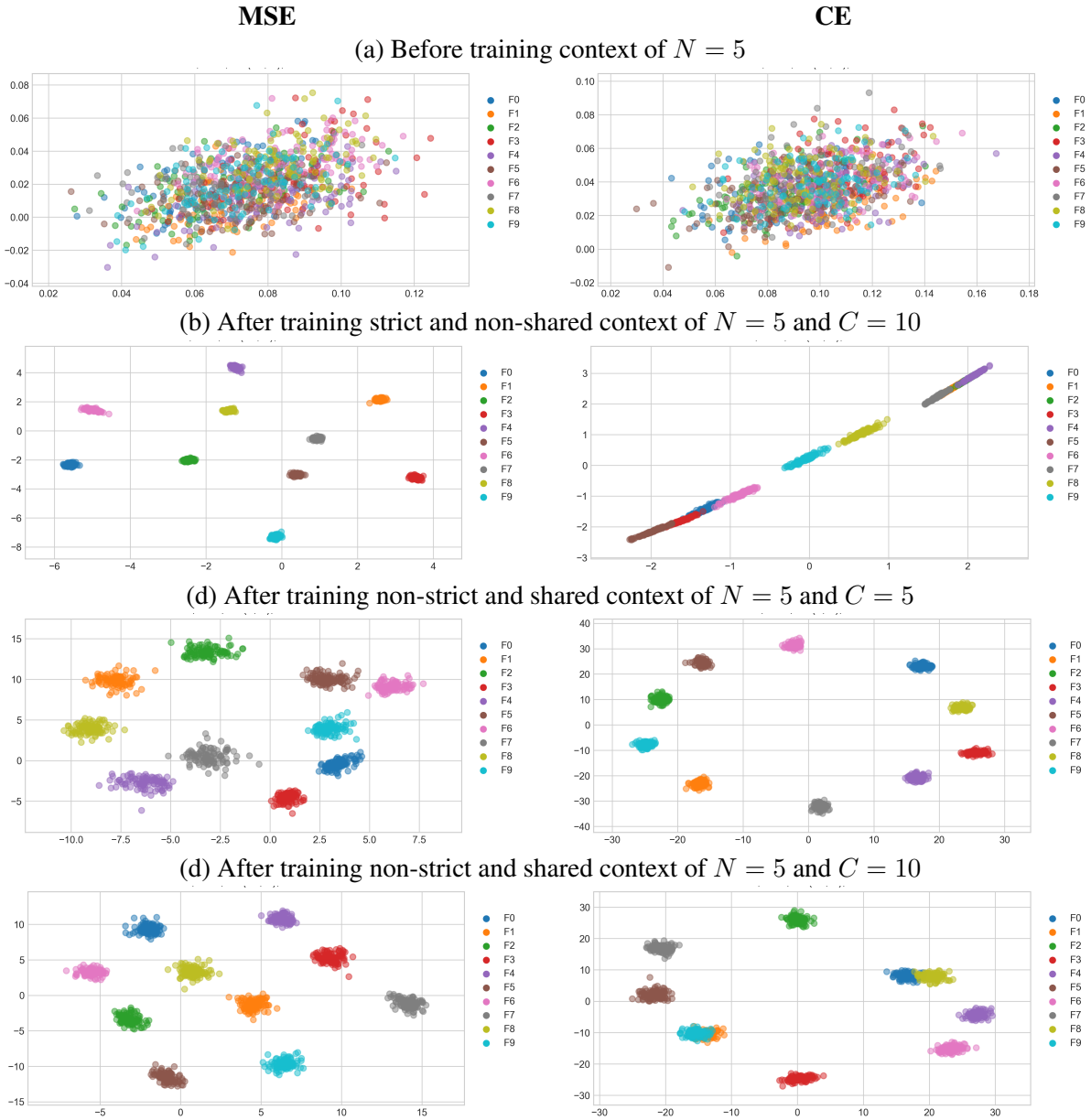

Figure 2: Messages sampled from latent space under different training and context settings. Colors represent the $f_i \in F$ input part of Sender.

5 Conclusion and Future Work

In addition to the previous work, we perform the same experiment under the two different loss setting and compare messages analytically and visually. We find point reflection under the CE setting and think it positively correlates the messages of more compositional than that under the MSE setting. The composition network is likely not powerful enough to capture the compositionality under the CE setting. Therefore, we can vary the composition network structure to see if there is an improvement over the communicative success by composition. We also find line alignment under the CE setting deteriorates the discreteness. Plus, our results are experimental, and we don't have a theoretical work supporting the phenomenon of line alignment and point reflection. Therefore, a valid explanation or further experimental analysis should be another part of future work.

References

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. 2013. Estimating or Propagating Gradients Through Stochastic Neurons for Conditional Computation.

Rahma Chaabouni, Eugene Kharitonov, Emmanuel Dupoux, and Marco Baroni. 2019. Anti-efficient encoding in emergent communication. In *Proceedings of the 33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*.

Dawn Chen, Joshua C. Peterson, and Thomas L. Griffiths. 2017. Evaluating vector-space models of analogy. In *Proceedings of the 39th Annual Conference of the Cognitive Science Society*.

Noam Chomsky. 1957. Logical structures in language. *American Documentation*, 8(4):284–291.

Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*, volume 96, pages 226–231.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep Sparse Rectifier Neural Networks. In *14th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 315–323.

Serhii Havrylov and Ivan Titov. 2017. Emergence of Language with Multi-agent Games: Learning to Communicate with Sequences of Symbols. In *Proceedings of the 31st Conference on Neural Information Processing Systems (NeurIPS 2017)*.

Dieuwke Hupkes, Verna Dankers, Mathijs Mul, and Elia Bruni. 2020. Compositionality decomposed: how do neural networks generalise?

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *International Conference of Learning Representations (ICLR)*.

Brenden M. Lake and Marco Baroni. 2017. Still not systematic after all these years: On the compositional skills of sequence-to-sequence recurrent networks. *CoRR*, abs/1711.00350.

Nur Geffen Lan, Emmanuel Chemla, and Shane Steinert-Threlkeld. 2020. On the spontaneous emergence of discrete and compositional signals.

Angeliki Lazaridou, Karl Moritz Hermann, Karl Tuyls, and Stephen Clark. 2018. Emergence of Linguistic Communication from Referential Games with Symbolic and Pixel Input. In *International Conference of Learning Representations (ICLR 2018)*.

Angeliki Lazaridou, Alexander Peysakhovich, and Marco Baroni. 2017. Multi-Agent Cooperation and the Emergence of (Natural) Language. In *International Conference of Learning Representations (ICLR2017)*.

David Lewis. 1969. *Convention*. Blackwell.

Tal Linzen. 2016. Issues in evaluating semantic spaces using word analogies. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 13–18, Berlin, Germany. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *arXiv:1301.3781 [cs]*. ArXiv: 1301.3781.

Richard Montague. 1970. Universal grammar. *Theoria*, 36(3):373–398.

Vinod Nair and Geoffrey E Hinton. 2010. Rectified Linear Units Improve Restricted Boltzmann Machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*.

S Pinker. 1996. *Language Learnability and Language Development (1984/1996)*. Cambridge, MA: Harvard University Press.

John Schulman, Nicolas Heess, Theophane Weber, and Pieter Abbeel. 2015. Gradient Estimation Using Stochastic Computation Graphs. In *Advances in Neural Information Processing Systems 28 (NIPS 2015)*.

Brian Skyrms. 2010. *Signals: Evolution, Learning, and Information*. Oxford University Press.

Shane Steinert-Threlkeld. 2018. Paying Attention to Function Words. In *Emergent Communication Workshop @ NeurIPS 2018*.

Shane Steinert-Threlkeld. 2020. Towards the Emergence of Non-trivial Compositionality. *Philosophy of Science*.

Elias C.G. Thijsse. 1987. Edward l. keenan amp; leonard m. faltz. 1985. boolean semantics for natural language. *Studies in Language. International Journal sponsored by the Foundation "Foundations of Language"*, 11(1):235–244.