Emergence of Compositional Public Languages from Multiple Agent's Private Languages in Neural Networks

Tolgahan Toy

***Abstract*—This paper proposes a new approach to emergent communication through neural networks by studying the development of a compositional, public language from multiple agents' private languages. The individual agents are trained to generate their private languages by producing continuous output states, which are not discretized. These output states are encoded and simulated to generate hidden states from the original images. After training, we combine the private languages of multiple agents into a common pool and create a new language in a supervised manner.**

To measure compositionality, we introduce a new technique based on the premise that similar expressions should have corresponding meanings. We frame a movie into several images in temporal order, considering images around the same time to be similar. The emergent language's compositionality degree is measured by analyzing the relationship between similar messages and their corresponding images.

The proposed approach offers a novel way of studying the emergence of a compositional language through neural networks. It provides insights into how agents can develop their private languages and how a common language can emerge from multiple private languages. The technique for measuring compositionality offers a way to evaluate the effectiveness of the emergent language.

I. Introduction

Emergent communication involves the spontaneous development of communication systems within a group of individuals without a pre-established language. This phenomenon is closely related to broader social behavior, particularly in terms of how individuals come together to establish and adhere to social norms. Historically, social contract theories have offered insights into this process. John Locke postulated that social rules emerge from the cooperation of self-interested agents ([1]), while Thomas Hobbes argued that political norms arise from competition among self-interested individuals ([2]). Contrasting with both, Jean-Jacques Rousseau posited that social norms cannot be reduced to self-interested individuals, as they are underpinned by the "general will" ([3]).

Similarly, distinct approaches exist regarding the emergence of communication. One perspective suggests that communication originates from the cooperation of self-interested individuals, while another maintains that it arises in a competitive environment. Both approaches, however, attribute the process to self-interested agents. Emergent communication through reinforcement learning exemplifies this perspective, as agents are either punished or rewarded, leading to behavioral adjustments.

In reinforcement-based emergent communication models, a neural agent responds to environmental input and generates a message for another agent. The second agent decodes the message and acts accordingly. Both agents update their parameters to maximize rewards, driven by self-interest.

An alternative approach to emergent communication transcends the reduction to self-interested individuals. Echoing Rousseau's concept of "general will," Margaret Gilbert's notion of the "imagined We" represents a social object that cannot be reduced to isolated individuals ([4]) ([5]). In this paper, we propose a model inspired by this perspective. In our model, agents do not develop a public language through self-interested interactions; instead, they regularize a pool of individual behaviors.

II. Related Work

[6] [7] [8] [9] [10] [11] [12] [13] [14] [15]

III. Private Language Construction in Neural Networks

In the emergent communication setting that we propose, the initial step involves each agent developing their own private language, which can be likened to an autoencoder. When an agent receives environmental input, it triggers a specific state within the agent. This state is then decoded into a set of symbols, and when these symbols are subsequently encoded, the agent returns to a similar state as it was when exposed to the original input.

This process bears resemblance to the distinction between primary and secondary language games proposed by Wittgensteinian philosophers Merrill B. Hintikka and Jaakko Hintikka. In a primary language game, an agent naturally communicates their sensations, such as a baby expressing pain through natural expressions. Movements in primary language games are not subject to correction, as the agent is not responsible to any community and does not need to adhere to social norms. Secondary language games, on the other hand, arise from interactions between various language games [16] [17].

In order to implement this approach, an agent is presented with an image as input. This input is processed by a CNN, and the resulting output is forwarded to a decoder. The decoder generates continuous outputs, which are then fed into an encoder that produces a hidden state (as illustrated in Figure 1). The following sections will provide a summary of each component in this architecture.

In our proposed architecture, we process input images of size $32 \times 32 \times 3$ and generate a high-dimensional hidden state through a CNN to be sent to the decoder. The CNN consists of 3 convolutional layers followed by batch normalization and ReLU activation functions. The first convolutional layer has 32 filters of size 3×3 with a stride of 1 and 'same' padding, followed by a second convolutional layer with 64 filters of size 3×3, also with a stride of 1 and 'same' padding. Max pooling with a 2×2 kernel and stride of 2 is applied after the second convolutional layer. The third convolutional layer has 128 and 512 filters with 3×3 kernel sizes, a stride of 1, and 'same' padding. The network then employs a global average pooling layer to reduce spatial dimensions before passing the output through a fully connected layer with 256 units and a ReLU activation function. Dropout with a rate of 0.5 is applied after the fully connected layer to reduce overfitting. The final output layer is a fully connected layer with the number of units equal to the size of the hidden state that is going to be generated.

The decoder in our proposed architecture is a GRU (Gated Recurrent Unit) designed to process the high-dimensional hidden state generated by the CNN and output continuous vectors. The GRU decoder consists of a single layer with 512 hidden units. The initial hidden state of the GRU is obtained from the high-dimensional hidden state produced by the CNN. At each time step, the GRU outputs a continuous vector with a dimensionality appropriate for the subsequent encoder. A linear activation function is applied at the output layer to ensure that the generated vectors can take on any real value.

Similarly, the encoder is also a GRU-based network that processes the continuous output vectors generated by the decoder. The encoder comprises a single layer with 512 hidden units. It receives the continuous output vectors from the decoder and generates a final hidden state, h_{end} at the end of the sequence. During training, the objective is to minimize the distance between the initial hidden state generated by the CNN, h_1, and the final hidden state generated by the encoder, h_{end}. The distance between these hidden states is measured using cosine similarity as the loss function. The optimization strategy for the entire architecture, including the CNN, decoder, and encoder, is Stochastic Gradient Descent (SGD) with a learning rate of 0.0001 and momentum of 0.9.

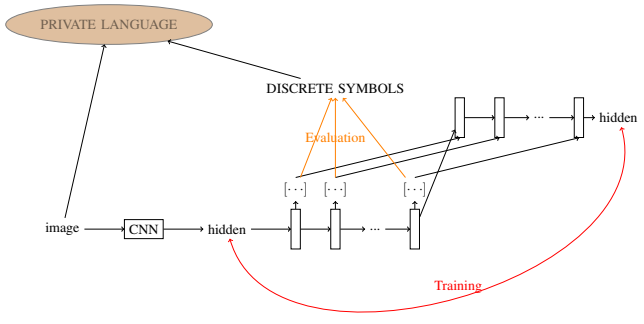

Fig. 1. Private Language

IV. Forming a Public Language from Private Languages

In the first part, we generated multiple private languages, one for each agent. Now we have a community of agents, each with their own unique private language. Agents all share the same environment and produce sequences of symbols in response to sensory input. In the end, we obtain a pool of sensory input-expression pairs. In our proposal, the community as a whole seeks to reduce this pool to a set of rules. Based on these rules, new data consisting of image-expression pairs for each image is generated. Each agent then learns from these pairs in a supervised manner.

While each agent is considered a neural network, the community that regularizes agents' behaviors does not function like one. Neural networks are complex, continuous, and probabilistic mechanisms, whereas community rules should be simpler, discrete, and logical. The agents are psychological entities, while the community is epistemological. Building on Daniel Kahneman's distinction between two cognitive systems, the agents' behavior is typically associated with System 1 thinking, while community rules are considered to reflect System 2 thinking ([18]).

To model community rules, we propose the use of Tsetlin Machines. These machines are an alternative to neural networks introduced by Ole-Christoffer Granmo and inspired by Michael Lvovtich Tsetlin's learning automata ([19]). Tsetlin Machines employ a set of Boolean inputs to produce Boolean outputs. In the case of multiclass machines, each output class follows the same structure. A Tsetlin Machine consists of several Automata, with each Boolean input and its negation sent to an automaton. Each automaton ultimately reaches either an "include" or "exclude" state, which determines whether to add or subtract 1 from the relevant category with its respective polarity. The final output is determined by comparing the difference between positive and negative polarity scores against a predetermined threshold.

Convolutional Tsetlin Machines, which employ shared automata, are designed to process images ([20]). When given an image vector in binary form, the machine outputs a Boolean value for the target category. In our case, the machine generates a set of values that represent various properties of linguistic symbol sequences. We train the Convolutional Tsetlin Machine to fit the data collected from private agents (as illustrated in Figure 2).

Once the Convolutional Tsetlin Machine has been trained, it can process new images from the dataset and output a set of expressions. We then construct a new dataset that consists of images and expressions generated by the trained Convolutional Tsetlin Machine (see Figure 3). Each agent, equipped with their own CNN and decoder, is trained in a supervised manner using this new dataset (as illustrated in Figure 4). Eventually, each agent possesses a language that initially developed privately but ultimately conforms to community standards. We can repeat this process as needed.

V. Measuring the Compositionality

The compositionality principle states that the meaning of an expression is a function of its constituents, and it defines

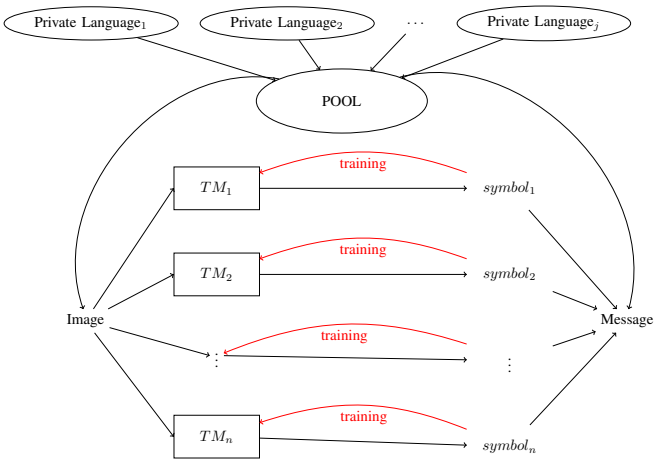

Fig. 2. Regularizing private languages

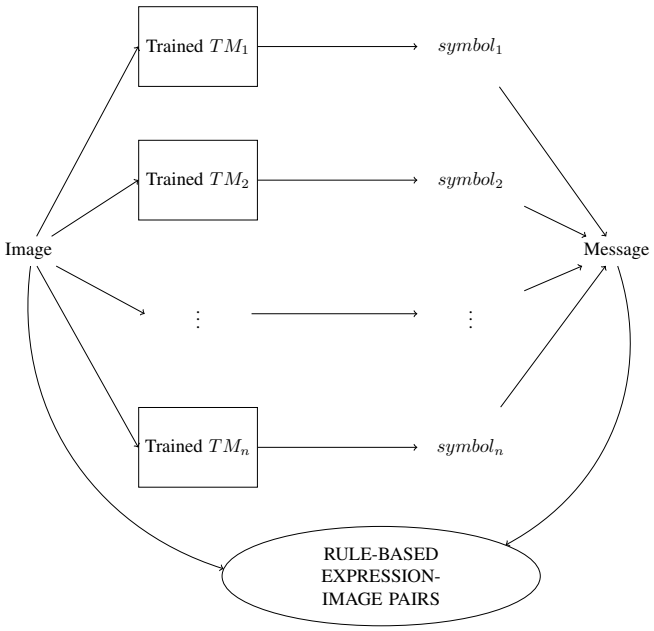

Fig. 3. Generating new data with social rules

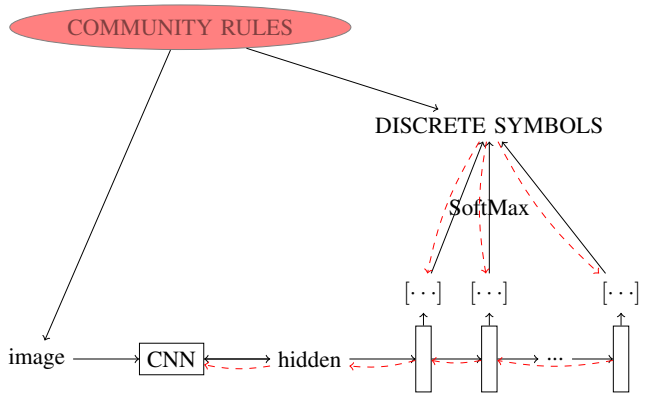

Fig. 4. Agents are supervised by community rules

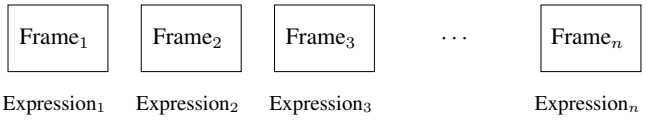

Fig. 5. To frame a video into a temporal sequence, we associate each image with an expression generated by an agent. We assume that frames with similar numbers are similar images.

the meaning function, h, as a homomorphism from syntactic algebra, $< A, F >$ to semantic algebra, $< B, G >$.

$$h(F(\alpha_1, ..., \alpha_n)) = G(h(\alpha_1, ..., h(\alpha_n))$$

This principle leads to the supervenience thesis, which asserts that there can be no change in meaning without a change in the expression.

We can refine this thesis to the loose supervenience thesis, which suggests that there can be no big change in meaning without a big change in the expression. To measure the compositionality of a language, we check if similar expressions have similar meanings. To do this, we need to measure the similarity between expressions and meanings, which are represented as images. We can use a metric to compare image similarity by assuming that images shown around the same time in a video are similar. As depicted in Figure 5, we frame the video into a temporal sequence of images and associate each image with an expression in the public language.

If two expressions are similar, we expect corresponding images shown around the same time to be similar too. We add +1 to the number of compositional expressions if the corresponding images are similar. The compositionality score is computed by dividing the number of similar expression pairs with corresponding images seen around the same time by the total number of similar expression pairs.

Algorithm 1 Measuring the compositionality

procedure MEASURE COMPOSITIONALITY
  $compositional \leftarrow 0$
  for $a, b < Expressions$ and $m(a) = Frame_\alpha, m(b) = Frame_\beta$ **do**
    if $a \approx b$ **then**
      if $\alpha - \beta < 10$ **then**
        $compositional \leftarrow compositional + 1$
      end if
    end if
  end for
  return a
end procedure

REFERENCES

[1] J. Locke, *Two Treatises of Government*. New York: Cambridge University Press, 1960, ch. Second Treatise, pp. 265–428.

[2] T. Hobbes, *Leviathan*. Indianapolis: Hackett Publishing Company, 1994, ch. XIII Of the Natural Condition of Manking, As Concerning Their Felicity, and Misery, pp. 74–78.

[3] J.-J. Rousseau, *The Social Contract A new translation by Christopher Betts*. Oxford: Oxford University Press, 1994, ch. VI The Social Pact, pp. 54–56.

[4] M. Gilbert, *A Theory of Political Obligation*. Oxford: Clarendon Press, 2006.

[5] ——, *Joint Commitment How We Make the Social World*. Oxford: Oxford University Press, 2014.

[6] A. Wong, T. Bäck, A. V. Kononova, and A. Plaat, "Deep multiagent reinforcement learning: challenges and directions," *Artificial Intelligence Review*, 2022.

[7] J. Russin, R. Fernandez, H. Palangi, E. Rosen, N. Jojic, P. Smolensky, and J. Gao, "Compositional processing emerges in neural networks solving math problems," *CoRR*, vol. abs/2105.08961, 2021. [Online]. Available: https://arxiv.org/abs/2105.08961

[8] R. Chaabouni, E. Kharitonov, D. Bouchacourt, E. Dupoux, and M. Baroni, "Compositionality and generalization in emergent languages," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, Jul. 2020, pp. 4427–4442. [Online]. Available: https://aclanthology.org/2020.acl-main.407

[9] I. Mordatch and P. Abbeel, "Emergence of grounded compositional language in multi-agent populations," *CoRR*, vol. abs/1703.04908, 2017. [Online]. Available: http://arxiv.org/abs/1703.04908

[10] M. Noukhovitch, T. LaCroix, A. Lazaridou, and A. C. Courville, "Emergent communication under competition," *CoRR*, vol. abs/2101.10276, 2021. [Online]. Available: https://arxiv.org/abs/2101.10276

[11] M. Baroni, R. Dessi, and A. Lazaridou, "Emergent language-based coordination in deep multi-agent systems," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: Tutorial Abstracts*. Abu Dubai, UAE: Association for Computational Linguistics, Dec. 2022, pp. 11–16. [Online]. Available: https://aclanthology.org/2022.emnlp-tutorials.3

[12] A. Lazaridou and M. Baroni, "Emergent multi-agent communication in the deep learning era," *CoRR*, vol. abs/2006.02419, 2020. [Online]. Available: https://arxiv.org/abs/2006.02419

[13] J. N. Foerster, Y. M. Assael, N. de Freitas, and S. Whiteson, "Learning to communicate with deep multi-agent reinforcement learning," *CoRR*, vol. abs/1605.06676, 2016. [Online]. Available: http://arxiv.org/abs/1605.06676

[14] S. Havrylov and I. Titov, "Emergence of language with multi-agent games: Learning to communicate with sequences of symbols," *CoRR*, vol. abs/1705.11192, 2017. [Online]. Available: http://arxiv.org/abs/1705.11192

[15] L. Galke, Y. Ram, and L. Raviv, "What makes a language easy to deep-learn?" 2023.

[16] J. H. Merrill B. Hintikka, *Investigating Wittgenstein*. Oxford: Basil Blackwell, 1986, ch. Differences and Interrelations among Language-games in wittgenstein, pp. 272–304.

[17] J. Hintikka, *Ludwig Wittgenstein: Half-Truths and One-and-a-Half-Truths*. Dordrecht: Kluwer Academic Publishers, 1996, ch. (with Merrill B. Hintikka) Different Language-Games in Wittgenstein, pp. 335–344.

[18] D. Kahneman, *Thinking, Fast and Slow*. New York: Farrar, Straus and Giroux, 2011.

[19] O. Granmo, "The tsetlin machine - A game theoretic bandit driven approach to optimal pattern recognition with propositional logic," *CoRR*, vol. abs/1804.01508, 2018. [Online]. Available: http://arxiv.org/abs/1804.01508

[20] O. Granmo, S. Glimsdal, L. Jiao, M. Goodwin, C. W. Omlin, and G. T. Berge, "The convolutional tsetlin machine," *CoRR*, vol. abs/1905.09688, 2019. [Online]. Available: http://arxiv.org/abs/1905.09688