ON SCALING AND MODELS OF LANGUAGE EVOLUTION

REMI VAN TRIJP[*1]

[*]Corresponding Author: remi.vantrijp@sony.com
[1]Sony Computer Science Laboratories Paris, Paris, France

Computational models of language evolution offer important insights for explaining the emergence and evolution of human languages. However, such models have recently been criticized for being computationally intractable. The goal of this paper is to show that this criticism is misleading because it reduces all models of language evolution to only a specific subset of models that assume that the basic unit of cultural transmission is the language itself, which leads to astronomically large hypothesis spaces. In fact, there is already decades worth of computational modelling using the Language Game paradigm that has successfully addressed the issue of scaling by treating language as a complex adaptive system that spontaneously evolves as the side-effect of local communicative interactions. This paper explains why the Language Game method scales so well, and how it incorporates insights from constructivist usage-based learning and Relevance theory. It will illustrate the method through a Naming Game, supported by open-source code that readers can download, test and reuse for their own work.

1. Introduction

Computational models of language evolution have played an important role in exploring the origins and evolution of human languages ever since the late 1980s and early 1990s (e.g. Hurford, 1989; Steels, 1995). However, a recent complexity analysis by Woensdregt et al. (2021) suggests that models of language evolution, at least in their current formulation, are computationally intractable so they cannot be scaled up to more ecological scales involving tens of thousands of words.

While complexity analysis can offer useful insights for scaling a model, the criticism of Woensdregt et al. (2021) is misleading because it reduces all models of language evolution to one specific kind of iterated model based on Bayesian inference (e.g. Griffiths & Kalish, 2007). Other kinds of models are not only ignored, but also simply discarded based on the following two arguments:

> "[I]t is not clear that other models [...] would not run into the same wall of intractability. Moreover, the Bayesian formalism has the virtue of being able to model agents' epistemic states and transitions while remaining agnostic about the precise implementing mechanisms" (Woensdregt et al., 2021 p. 6).

The goal of this paper is to refute this conclusion by showing that there already exists decades worth of research using the Language Game methodology

(Steels, 1995, 2000) which has successfully addressed issues of scale by treating language as a complex adaptive system, and by drawing inspiration from constructivist usage-based language learning (Bybee, 2006; Goldberg, 2011) and Relevance Theory (Sperber & Wilson, 1986). This paper will explain why the Language Game method scales so well through a Naming Game (Steels, 1995; Baronchelli, Felici, Loreto, Caglioti, & Steels, 2006). Readers who are interested in running the Naming Game on their own computers can download the paper's supporting code for free as open-source software,[1] as well as the open-source framework Babel2 (Loetzsch, Wellens, De Beule, Bleys, & van Trijp, 2008) that has been used for the implementation, which can be downloaded at `https://gitlab.ai.vub.ac.be/ehai/babel-core`.

2. Illustrating the Problem of Scalability

Before turning to language games, it is important to understand the argument of computational intractability. Woensdregt et al. (2021) assume a Bayesian iterated learning model of cultural transmission in the style of Griffiths and Kalish (2007), in which an adult language user produces a number of utterances that are observed by a child learner, who forms hypotheses about which language could produce such utterances. Each hypothesis h is a language, so the *hypothesis space* consists of all of the possible languages. At the end of a cycle, there is a generational turnover in which the child becomes the adult and a new learner is introduced.

Woensdregt et al. (2021) provide a complexity analysis that shows that such models are computationally intractable. They illustrate the idea with the following example: suppose that a language is a set of one-to-one mappings between *signals* and *referents*, then the hypothesis space of all possible languages consists of all possible signal-referent mappings, which amounts to $2^{\#signals \times \#referents}$. Woensdregt et al. (2021) write that even for a toy language in which 50 signals exist for 25 referents, "learners need to consider all $2^{50\times25}$, about 1.9×10^{376}, possible languages" (p. 1). Increasing the number of signals and referents thus leads to a combinatorial explosion in the hypothesis space, which makes scaling impossible. In other words, such a model cannot be salvaged by faster computers or better implementations, because they face "a deeper theoretical issue" (p. 1).

Woensdregt et al. (2021) do not identify what exactly that deeper theoretical is, but the culprit seems obvious: the learner needs to consider an astronomically large hypothesis space, which is due to the fact that they have to consider the probabilities of all possible languages. This is a side-effect of the model's implicit assumption that the basic unit of cultural transmission is the language itself, as opposed to utterance-based models of cultural transmission (e.g. Croft, 2000) that make the learning task much more manageable.

[1] https://github.com/SonyCSLParis/Naming-Game

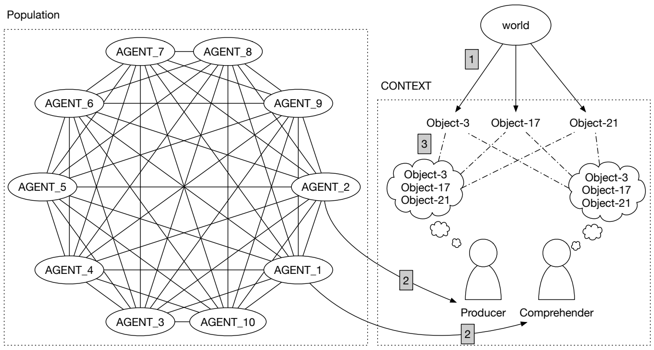

Figure 1. In the Language Game paradigm, two agents are drawn from a population of peers in order to engage in a locally situated communicative interaction.

3. The Naming Game

Let us now turn to the Language Game methodology, which is illustrated in Figure 1. A language game experiment typically involves a population of multiple agents, organized according to a network topology that represents the population's social structure. At each *time step* of the experiment, two agents are drawn from the population to play a *language game* with each other, which is a locally situated interaction that is private to the participating agents. Since all agents are peers, each agent can take on the role of producer or comprehender. During a language game, agents only worry about achieving their communicative goals with respect to the current situation, so they are not preoccupied by learning "the" community language. In fact, as will be demonstrated below, the community language spontaneously emerges and evolves through *self-organization* in very much the same way an ant path or other complex systems are formed in nature.

The simplest Language Game experiment is the Naming Game, which was first introduced by Steels (1995) and which has well-understood mathematical properties (Baronchelli et al., 2006; De Vylder & Tuyls, 2006). The Naming Game involves a population of N agents that need to self-organize a shared lexicon $\mathcal{L}$ for referring to a number M of objects present in their world. Each agent is endowed with an associative lexicon L_a that consists of a list of *lexical constructions*, here operationalized as signal-referent associations (in order to be consistent with the example of Woensdregt et al., 2021) that are assigned a *preference score* that represents the strength of an association. More formally: $L_a = \{\langle s_1, r_1, \sigma_1 \rangle, ...\}$ where s_i is a possible signal (a string), r_i a possible referent (a unique symbol), and where $0.0 \leq \sigma_i \leq 1.0$ is the preference score with as initial value 0.5. All agents start with an empty lexicon at time step $t = 0$, but gradually invent and learn new constructions as they interact with each other according to the following scenario:

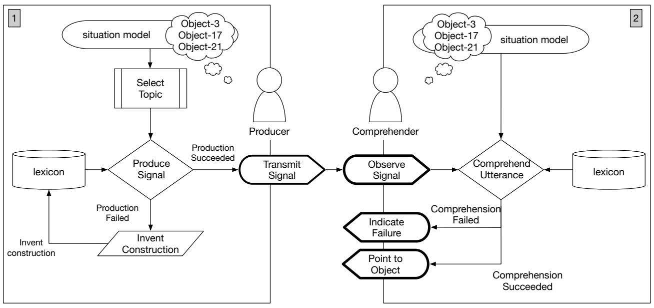

Figure 2. This Figure shows the mental operations that the producer goes through in order to produce an utterance, and the operations that the comprehender performs to comprehend the utterance.

1. Each time step t, a *situated context* is initialized, as illustrated in Figure 1.
   - (a) A context consists of n objects randomly selected from the "world."
   - (b) One agent is randomly selected to act as the *producer*, and a *comprehender* is selected among the agents that are directly connected to the producer's social network (shown on the left of Figure 1).
2. The agents start interacting. Figure 2 illustrates their linguistic actions:
   - (a) Both agents maintain a *situation model* of the current context, in which they keep track of the objects they perceive.
   - (b) The producer randomly selects one of the objects as the *topic*.
   - (c) The producer transmits a signal for referring to the topic to the comprehender:
     - If the producer does not know an appropriate signal yet for the topic, they will invent one. Here, they will randomly generate a string according to the template "CVCVCVCVCV" (e.g. "kebekobola");
     - If the producer knows more than one signal that associated with the topic ("competitors"), they will choose one according to the *inventory dynamics* of the experiment. In this paper, the agent chooses the construction that has the highest preference score.
   - (a) The comprehender tries to comprehend the producer's signal.
     - If the comprehender knows a construction that maps the signal onto one of the referents in their situation model, comprehension succeeded. They will point to that object.
     - If not, the comprehender will indicate failure to the producer.

3. The producer gives feedback to the comprehender. They will signal success if the comprehender pointed to the correct object, and signal failure otherwise. In the latter case they also point to the intended topic, so the the comprehender can *learn* a new signal-referent association.

4. The agents update their linguistic inventories based on the success or failure of the game according to the experiment's inventory dynamics.

As can be inferred from the above, the *inventory dynamics* of an experiment determine how the agents cope with variation in the population and how they update their linguistic inventories after each usage event (see Baronchelli, 2018 for a primer on suitable strategies for achieving consensus). In the experiments of this paper, agents update their inventories using *lateral inhibition*, which means that they will increase the preference scores of lexical constructions that led to success while at the same time punishing competitors by lowering their preference scores, which is compatible with more recent proposals on statistical preemption in constructivist language learning (Goldberg, 2011). Agents will also punish constructions that led to communicative failure. The experiments in this paper adopt the score updating rule of De Beule, De Vylder, and Belpaeme (2006), which has been proven successful in prior research.

4. Experimental Results and Discussion

To remain close to the example of Woensdregt et al. (2021), a Naming Game experiment was set up with a population size of $N = 10$ and a number of objects $M = 25$. In order to test whether the population succeeds at self-organizing a lexicon for referring to these 25 objects, 100 independent simulations were executed with each 6.000 time steps, which amounts to an average of 1.200 interactions per agent per simulation.

Figure 3 shows the most important results averaged over the hundred simulations, with error bars indicating the variability between each run. The measure Communicative Success is a running average of the past 10 interactions in which a failed game counts as 0 and a successful game as 1. As can be seen, the agents already reach success after about 2.000 time steps (about 400 interactions per agent or 16 interactions per object). Communicative success however doesn't mean that agents have reached *consensus* about which signal to use for which referent, because each agent might simply have learned all of the signals in order to understand the others, but keep using their own preferences. The consensus of a signal-referent association is measured as the inverse of normalized (Shannon) entropy, as formally defined in the supporting code. Global consensus simply averages over these individual consensus scores. As can be seen in the left graph of Figure 3, consensus quickly follows communicative success, with maximal consensus after about 4.000 interactions, an average of 800 interactions per agent, or 32 interactions per referent.

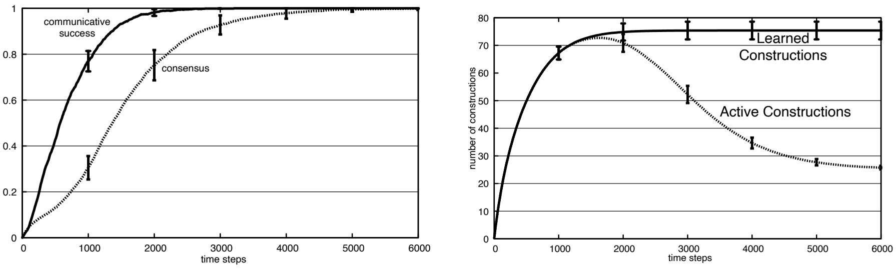

Figure 3. Naming Game with $N = 10$ and $M = 25$. Results average over 100 simulations. Left: Agents reach persistent communicative success after about 2.000 time steps and a global consensus after about 5.000 time steps. Right: Agents have to learn on average 75 words, but their alignment strategy allows them to reduce their active lexicon to an optimal size of one signal per referent.

The graph on the right shows more information about the emergent lexicon. Each agent learns about 75 different constructions for referring to 25 objects. However, through lateral inhibition the agents succeed in reducing their lexicons to an optimal size of 25 active constructions. Constructions are considered to be active as long as their preference score exceeds a threshold $\sigma_i \geq 0.2$, otherwise they become dormant. Hidden from the graph is the actual variation in the population: every time a producer communicates about an object for the first time, they invent a new construction. Given that the producer is randomly selected, on average half of the population $N/2$ will invent a competitor for the same object, leading to an average of five competitors per referent or a total of 125 words for 25 referents circulating in the population. If the agents would have to consider the space of all possible languages, as in the formalization of Woensdregt et al. (2021), they would have to entertain $2^{125 \times 25}$, or about 5.2×10^{939}, possible languages. Yet, as readers who test the code can verify, it takes only a couple of seconds to run the simulations on a present-day laptop (in fact, it takes more time to produce the graphs than running the simulations themselves).

5. Discussion and Conclusion

The previous sections explained a simple Naming Game, but how does it scale? Prior research has provided mathematical proof that the Naming Game always converges (De Vylder & Tuyls, 2006), and Baronchelli et al. (2006) and Baronchelli (2006) have examined how the model behaves when scaling the population size, reporting simulations up to $N = 100000$. These experiments reveal that the model displays similar behavior as natural language dynamics, most notably that there is an S-shaped curve with sharp transition towards population-wide convergence. In fact, the transition becomes sharper and sharper as the size of the language increases. Baronchelli et al. (2006) conclude that this "surprising result [...] explains why human language can scale up to very large populations."

Other experiments have shown how the Language Game method can also successfully apply to more complex languages where there is no one-to-one mapping between signal and referent, including large lexicons and meaning spaces (Wellens, Loetzsch, & Steels, 2008), or experiments on grammatical structures such as argument structure constructions (see Steels, 2012 for a collection of experiments, and the open-access book series *Computational Models of Language Evolution* at Language Science Press).

Why doesn't the Language Game method hit the wall of computational intractability? The answer is that the agents never exhaustively search the hypothesis space, but instead only consider what is *relevant* for achieving communicative success in their local interactions. For example, when a learner observes a particular signal, they will not try to update their entire lexicon, but only those constructions that were involved in the language game: the constructions that were used and their competitors. In the current setup, an agent knows on average three competitors for one referent, so they will on average never make more than three local adjustments (i.e. update their preference scores) at each time step instead of recalculating the probabilities of all possible mappings. Global consensus is nevertheless achieved as a side-effect in the same way as ant paths are spontaneously formed as the side-effect of local behaviour. Language can therefore be seen as a *complex adaptive system* that is constantly shaped and reshaped through language usage (Steels, 2000).

In sum, when Woensdregt et al. (2021) posed the challenge of scaling, they wrongly equated "models of language evolution" with a particular kind of model in which a language must be learned as a whole. Other models that operationalize language learning as a much more manageable task, such as utterance-based models of language evolution (Croft, 2000) or the Language Game paradigm, do not run into the problem of computational intractability; and have in the latter case already been demonstrated to scale to more realistic settings. And just like the Bayesian iterated learning models, the Language Game method is a general framework that is agnostic to the specifics of implementation, as can be gleaned from the breadth of techniques and phenomena that have already been investigated with this method (Steels, 2012).

Acknowledgements

The work reported in this paper has been funded by the Sony Computer Science Laboratories Paris. The author wishes to thank Luc Steels for his invaluable feedback, as well as Martina Galletti and Inès Blin, and the many researchers who contributed through discussions and collaborations on the Babel software suite, particularly Katrien Beuls (Université de Namur) and Paul Van Eecke (VUB – KULeuven). Special thanks also go to Hiroaki Kitano, Vittorio Loreto and Peter Hanappe for creating such a superb work environment.

References

Baronchelli, A. (2006). *Statistical mechanics approach to language games.* Unpublished doctoral dissertation, Università di Roma "La Sapienza", Rome.

Baronchelli, A. (2018). The emergence of consensus: A primer. *Royal Society Open Science*, *5*(172189).

Baronchelli, A., Felici, M., Loreto, V., Caglioti, E., & Steels, L. (2006). Sharp transition towards shared vocabularies in multi-agent systems. *Journal of Statistical Mechanics*, *P06014*.

Bybee, J. (2006). From usage to grammar: The mind's response to repetition. *Language*, *82*(4), 711–733.

Croft, W. (2000). *Explaining language change: An evolutionary approach.* Harlow, Essex: Longman.

De Beule, J., De Vylder, B., & Belpaeme, T. (2006). A cross-situational learning algorithm for damping homonymy in the guessing game. In L. M. Rocha, L. S. Yaeger, M. A. Bedau, D. Floreano, R. L. Goldstone, & A. Vespignani (Eds.), *Artificial life x* (pp. 466–472). MIT Press.

De Vylder, B., & Tuyls, K. (2006). How to reach linguistic consensus: A proof of convergence for the naming game. *Journal of Theoretical Biology*, *242*(4), 818–831.

Goldberg, A. E. (2011). Corpus evidence of the viability of statistical preemption. *Cognitive Linguistics*, *22*(1), 131–154.

Griffiths, T. L., & Kalish, M. L. (2007). Language evolution by iterated learning with Bayesian agents. *Cognitive Science*, *31*, 441–480.

Hurford, J. R. (1989). Biological evolution of the Saussurean sign as a component of the language acquisition device. *Lingua*, *77*(2), 187–222.

Loetzsch, M., Wellens, P., De Beule, J., Bleys, J., & van Trijp, R. (2008). *The babel2 manual* (Tech. Rep. No. AI-Memo 01-08). Brussels: AI-Lab VUB.

Sperber, D., & Wilson, D. (1986). *Relevance: Communication and cognition.* Cambridge, MA: Harvard University Press.

Steels, L. (1995). A self-organizing spatial vocabulary. *Artificial Life*, *2*(3), 319–332.

Steels, L. (2000). Language as a complex adaptive system. In M. Schoenauer, K. Deb, G. Rudolph, X. Yao, E. Lutton, J. Merelo, & H.-P. Schwefel (Eds.), *Parallel problem solving from nature* (pp. 17–28). Berlin: Springer-Verlag.

Steels, L. (Ed.). (2012). *Experiments in cultural language evolution.* Amsterdam: John Benjamins.

Wellens, P., Loetzsch, M., & Steels, L. (2008). Flexible word meaning in embodied agents. *Connection Science*, *20*(2–3), 173–191.

Woensdregt, M., Spike, M., Haan, R. de, Wareham, T., Rooij, I. van, & Blokpoel, M. (2021). *Why is scaling up models of language evolution hard?* PsyArXiv. (https://doi.org/10.31234/osf.io/d2h5c, version of May 2021)