An Agent-Based Model of the Emergence and Evolution of a Language System for Boolean Coordination*

JAAMAS Track

Josefina Sierra-Santibáñez
Technical University of Catalonia, Spain. Maria.Josefina.Sierra@upc.edu

ABSTRACT

This paper presents an agent-based model for studying the emergence and evolution of a language system Boolean coordination. The model has been implemented and tested by conducting a series of experiments that show that a language system for Boolean coordination emerges as a result of a process of self-organisation of the agents' linguistic interactions when these agents adapt their preferences for vocabulary, syntactic categories and word order to those they observe are used more often by other agents, and that such a language system is reliably transmitted across generations.

Keywords: Language Evolution; Agent-Based Model; Coordination

ACM Reference Format:
Josefina Sierra-Santibáñez. 2019. An Agent-Based Model of the Emergence and Evolution of a Language System for Boolean Coordination. In *Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), Montreal, Canada, May 13–17, 2019,* IFAAMAS, 3 pages.

1 INTRODUCTION

The question of the *origins and evolution of language* has been approached in AI through the construction of *agent-based models* [5, 8, 13, 17, 19, 26]. Such models involve a population of autonomous agents that interact with each other playing language games. A language game [30] is an interaction between two agents, a speaker and a hearer. The speaker has a communicative goal, conceptualises the world for language, transforms this conceptualisation[1] into an utterance, and communicates that utterance to the hearer. The hearer tries to parse the utterance, reconstruct its meaning and map it onto its own internal representation of the world. Speaker and hearer use extralinguistic means to determine the outcome of a language game[2] and, depending on that outcome, expand and refine their internal languages (e.g. updating the scores of their lexical entries to adapt their vocabulary and word order preferences).

The agents in these models are initially endowed with some cognitive abilities that are assumed to be necessary to take part in the particular language game studied. In the simulations, the agents are made to play a series of language games where they configure possible language systems and try them out. The goal of the experiments is to find out whether the population as a whole communicates effectively, and to observe the conceptualisations and linguistic constructions that emerge in the population as a result of the processes of collective invention and negotiation, as well as the evolution of several features of the emerging languages.

Theories of language evolution study language change at two different levels: that of language systems and that of language strategies. *Language systems* capture the regularity observed in some part of the vocabulary or the grammar of a language [25]. A language system groups a set of paradigmatic choices both on the side of meaning (the conceptual system) and on the side of form (the linguistic system). The *conceptual system* contains the semantic distinctions that are expressible in the language system and can therefore be used as building blocks for conceptualisation. The *linguistic system* includes the syntactic categories, lexicon and/or grammatical constructions necessary to turn a conceptualisation into a concrete utterance. Linguists call the approach underlying a language system a *language strategy*. Examples of language systems that have been studied using agent-based models are: (1) *case systems* [3, 24]; (2) *agreement markers* [4]; (4) *phrase-structure* [12, 29]; or (5) *vocabularies* co-evoling *semantic categories* [16, 27, 28].

In this paper we study the evolution of a language system for *coordination*, a linguistic phenomenon which involves combining syntactic categories using coordinators like *and, or*, in order to communicate logical combinations of certain concepts, for example:
a. Terry $[_{np\backslash s}$ [jumps] and [runs]].
b. John ran $[_{s\backslash s}$[in Pittsburgh yesterday] and [in Cleveland today]].
The fundamental principle of coordination is that any pair of categories of the same Boolean type can be coordinated to produce a result of the same category [7]. A Boolean type is one that eventually produces a Boolean category after applying to all of its arguments. Any category producing a nominal or sentential result is Boolean. Thus, nouns, sentences, adjectives, adverbs, prepositions, prepositional phrases and verbs, among others, can be coordinated.

The previous examples share the property that the coordinator appears as a Boolean operator that has been distributed past the coordinate categories. For example, *'Terry and Sandy studied'* can be read as *'Terry studied and Sandy studied'*, where the Boolean operator applies not only to the proper nouns but to the complete coordinate clauses. This form of coordination, known as *Boolean coordination*, is the type of coordination we will focus on[3].

[*] This paper is an extended abstract of an article published in Autonomous Agents and Multi-Agent Systems [23]. This work has been partially supported by funds from the Spanish Ministry for Economy and Competitiveness and the European Union under grants TIN2017-86727-C2-1-R, TIN2014-57226-P, 2017SGR786, and 2014SGR890.

[1] In the experiments described in this paper, a conceptualisation (i.e. a *meaning*) is a higher-order logic term that is true for all the objects in the subset of objects that constitutes the topic of the language game and false for the rest of the objects.

[2] A language game *succeeds* if the hearer can parse the expression communicated by the speaker and if its interpretation of that expression is *logically equivalent* to the meaning the speaker had in mind; otherwise, the language game *fails.*

[3] There is also a different type of coordination, called *non-Boolean coordination.* Examples of non-Boolean coordination are *'Broke is strong enough to carry Jody and Francis'*, or *'the professors and the dean met'*. To represent the meaning of these sentences, a scheme is needed to coordinate noun phrases to produce sets, which can then be interpreted collectively or distributively. Furthermore, such scheme must be able to account for examples coordinating singular and plural noun phrases, as in the examples.

Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), N. Agmon, M. E. Taylor, E. Elkind, M. Veloso (eds.), May 13–17, 2019, Montreal, Canada.

2 CONCEPTUAL SYSTEM

The formalism the agents use to build internal representations of the *subsets of objects* they try to communicate about is the *simply typed λ-calculus* [9], and specifically *higher-order logic* [18]. In particular, we assume the agents can build complex meanings such as `'the objects which are either up or to the left, but not both'`, which is a logical combination of basic properties and can be represented by the higher-order logic term[4] $\lambda x.and(or(up(x))(le(x)))(not(and(up(x))(le(x))))$.

3 LINGUISTIC SYSTEM

The grammatical formalism the agents use to represent the lexicons they construct is *categorial grammar*[5]. In *categorial grammar* every syntactic category corresponds to some higher-order type, with the assumption being that expressions of each category can be assigned meanings of the appropriate type. We assume a set of basic categories $Bascat = \{np, n, s, pr\}$, which abbreviate *noun phrase, noun, sentence* and *property*, and are associated with the higher-order types $Type(np) = Ind$, $Type(s) = Bool$ and $Type(n) = Type(pr) = Ind \rightarrow Bool$. *Bascat* is used to generate an infinite set of functional categories, each of which specifies (possibly complex) argument and result categories. A category B/A or $A\backslash B$ is said to be a *functor* category, and to have an argument category A and a result category B. A *functional category* of the form B/A is called a *forward functor* and looks for its argument A to the right; a *backward functor* $A\backslash B$ looks for its argument to the left.

The fundamental operation in *applicative categorial grammar* is the concatenation of an expression assigned to a functional category and an expression of its argument category to form an expression of its result category. The order of concatenation is determined by the functional category. For instance, $np\backslash s$ is the category assigned to both verb phrases and intransitive verbs: they look backward for a noun phrase to produce a sentence. The category of prenominal adjectives is n/n: they look forward for nouns to produce nouns.

We suppose that at the beginning of a simulation run the agents have a common lexicon for referring to basic properties. The agents' lexicons may initially contain lexical entries such as $sup \Rightarrow \texttt{up}{:}pr : 1.0$. A lexical entry $e \Rightarrow \alpha{:}A{:}s$ is an association (e, A, α, s) between a linguistic expression e, a syntactic category A, a higher-order logic term α, and a score s, such that α is a formal representation in higher-order logic of the meaning of e, A is its syntactic category, and s is the score of the association. The score of a lexical entry is a real number in the interval $[0.0, 1.0]$ that estimates its usefulness in previous communication[6].

4 INVENTION, ADOPTION, SIMPLIFICATION THROUGH ABSTRACTION AND REPAIR

Invention and adoption allow the agents to construct and learn associations between expressions and meanings. For example, when an agent acting as speaker cannot express a meaning F using its lexicon, it invents an expression E to communicate F, and constructs an association of the form $E \Rightarrow \mathsf{F} : C$, where C is the syntactic category[7] of expression E. Similarly, an agent acting as hearer who cannot parse the expression E communicated by the speaker using its lexicon also constructs an association $E \Rightarrow \mathsf{F}{:}C$, where E and F are the expression and the meaning used by the speaker. From these associations the agents construct new lexical entries that they incorporate to their lexicons and use in succeeding language games to generate and interpret other sentences. However, the agents do not build lexical entries directly from the associations $E \Rightarrow \mathsf{F}{:}C$ they construct during the invention or adoption steps of a language game. They apply two *simplification rules* (called *backward simplification* and *forward simplification*[8]) to such associations, to generalise them and remove from them those parts they can generate using lexical entries which already belong to their lexicons.

Forward and *backward simplification* [23] differ from the induction rules used in [21, 22] in applying *deduction* to infer the meanings and syntactic categories of *unknown subexpressions* from the meanings and syntactic categories of the subexpressions surrounding them in a given utterance[9], instead of inducing new grammar rules and inventing syntactic categories for unknown subexpressions. Simplification rules construct the meanings and syntactic categories of *unknown subexpressions* (i.e. expressions which cannot be parsed by known lexical entries) by applying functional abstraction at the semantic and syntactic levels of an association.

Finally, if an agent realises that it generates an ambiguous expression to a express a given meaning, it applies a *repair operation*[10] to the lexical entries it used to generate such an expression.

5 EXPERIMENTS

The agent-based model proposed in this paper has been implemented in Prolog [6, 10, 11] and tested by conducting a series of experiments which study the emergence of a language system and its transmission across generations for different population sizes and *turnover interval*[11] lengths. In each experiment, we monitor the evolution of four measures (*communicative success, lexical variability*[12]*, invention, adoption*), in order to analyse the population's global performance. For reasonable turnover interval lengths, we observe that full communicative success and null lexical variability are reached at the end of each turnover interval, and that the average number of adoptions per agent decreases in succeeding generations (see [23]). This can be explained by the fact that new agents learn a language already established in the population (i.e. the language that is transmitted from generation to generation), which uses fewer variations for expressing each Boolean operator.

[4] up and le are constants of type $Ind \rightarrow Bool$ denoting the properties of being in an upper or left position. $Bool$ and Ind are the types of Boolean values and individuals.

[5] A well known linguistic application of categorial grammar is *Boolean coordination*. Instances of Boolean coordination including traditional constituents and categories not usually analysed as constituents are formalised in *categorial grammar* [1, 2, 14, 15, 20].

[6] The score of a sentence or meaning generated using some lexical entries is computed multiplying the scores of such lexical entries. Scores are used to select the expression or meaning an agent prefers in a language game: 1) if an agent acting as speaker can generate several expressions to communicate a given meaning, it chooses the expression with the highest score; 2) if an agent acting as hearer can obtain several meanings by parsing an expression, it selects the meaning with the highest score.

[7] The syntactic category of the expressions the agents invent in the experiments is pr.

[8] See [23] for a formal definition of simplification rules and application examples.

[9] *Deduction* uses the Lambek Calculus' algebraic interpretation of operators / and \.

[10] *Repair operations* specialise lexical entries which are too general and can generate ambiguous expressions (see [23] for a formal definition and application examples).

[11] The agents in the population are divided into three groups: the elder, the adults and the young. Every t_r games (*the turnover interval length*), the elder are replaced with new agents which have no lexical entries for Boolean operators, the previous adults become the elder, the young the adults, and the new agents the younger generation. As a consequence, the population is completely renewed every $3 \cdot t_r$ games.

[12] Lexical variability captures the dissimilarity between the agents' internal languages (i.e. the lexicons or sets of preferred lexical entries stored by each individual agent).

REFERENCES

[1] K. Ajdukiewicz. 1935. Die Syntaktische Konnexitat. *Studia Philosophica* 1 (1935), 1–27.

[2] Y. Bar-Hillel. 1950. On Syntactical Categories. *Journal of Symbolic Logic* 15 (1950), 1–16.

[3] J. Batali. 1998. Computational Simulations of the Emergence of Grammar. In *Approaches to the Evolution of Language: Social and Cognitive Bases*. Cambridge Univ Press, 405–426.

[4] K. Beuls and L. Steels. 2013. Agent-Based Models of Strategies for the Emergence and Evolution of Grammatical Agreement. *PLoS ONE* 8, 3 (2013), e58960.

[5] T. Briscoe (Ed.). 2002. *Linguistic Evolution through Language Acquisition: Formal and Computational Models*. Cambridge University Press, Cambridge.

[6] F. Bueno, D. Cabeza, M. Carro, M. Hermenegildo, P. López-García, and G. Puebla. 1997. *The Ciao Prolog System. Reference Manual.* Technical Report CLIP3/97.1. School of Computer Science, Technical University of Madrid (UPM). Available from http://www.clip.dia.fi.upm.es/.

[7] B. Carpenter. 1997. *Type-Logical Semantics.* MIT Press.

[8] E.A. Cartmill, S. Roberts, H. Lyn, and H. Cornish (Eds.). 2014. *Evolution of Language, Proceedings of the Tenth International Conference EVOLANG.* World Scientific.

[9] A. Church. 1940. A formulation of a simple theory of types. *Journal of Symbolic Logic* 5 (1940), 56–68.

[10] W.F. Clocksin and C.S. Mellish (Eds.). 1996. *Programming in Prolog* (fourth ed.). Springer.

[11] A. Colmerauer, H. Kanoui, R. Pasero, and P. Roussel. 1973. *Un système de Communication Homme-machine en Francais, Research Report.* Technical Report. Groupe Intelligence Artificielle, Université Aix-Marseille II, France.

[12] E. Garcia-Casademont and L. Steels. 2016. Insight Grammar Learning. *Journal of Cognitive Science* 17, 1 (2016), 27–62.

[13] J. Hurford, M. Studdert-Kennedy, and C. Kight (Eds.). 1998. *Approaches to the Evolution of Language: Social and Cognitive Bases.* Edinburgh University Press.

[14] J. Lambek. 1958. The mathematics of sentence structure. *Amer. Math. Monthly* 65 (1958), 154–169.

[15] J. Lambek. 1961. On the calculus of syntactic types. In *Structure of Language and its Mathematical Aspects: Proceedings of Symposia in Applied Mathematics.* American Mathematical Society, 166–178.

[16] J. Lara and M. Alfonseca. 2000. Some Strategies for the Simulation of Vocabulary Agreement in Multi-Agents Communities. *Journal of Artificial Societies and Social Simulation* 3, 4 (2000). http://jasss.soc.surrey.ac.uk/3/4/2.html.

[17] C. Lyon, C. Nehaniv, and A. Cangelosi (Eds.). 2007. *Emergence of Language and Communication.* Springer-Verlag, Berlin.

[18] John McCarthy. 1960. Recursive Functions of Symbolic Expressions and Their Computation by Machine, Part I. *Commun. ACM* 3, 4 (1960), 184–195.

[19] JW. Minett and WSY. Wang (Eds.). 2005. *Language Acquisition, Change and Emergence: Essays in Evolutionary Linguistics.* City University of Hong Kong Press.

[20] G.V. Morrill. 2011. *Categorial Grammar: Logical Syntax, Semantics, and Processing.* Oxford University Press.

[21] J. Sierra-Santibáñez. 2014. An Agent-Based Model Studying the Acquisition of a Language System of Logical Constructions. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, AAAI-2014.* AAAI Press, 350–357.

[22] J. Sierra-Santibáñez. 2015. An Agent-Based Model of the Emergence and Transmission of a Language System for the Expression of Logical Combinations. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, AAAI-2015.* AAAI Press, 492–499.

[23] J. Sierra-Santibanez. 2018. An Agent-Based Model of the Emergence and Evolution of a Language System for Boolean Coordination. *Autonomous Agents and Multi-Agent Systems* 32 (2018), 417–458. Issue 4.

[24] L. Steels. 1998. The Origins of Syntax in Visually Grounded Robotic Agents. *Artificial Intelligence* 103(1-2) (1998), 133–156.

[25] L. Steels. 2011. Modeling the Cultural Evolution of Language. *Physics of Life Reviews* 8 (2011), 339–356.

[26] L. Steels (Ed.). 2012. *Experiments in Cultural Language Evolution.* John Benjamins.

[27] L. Steels. 2015. *The Talking Heads experiment: Origins of words and meanings (Computational Models of Language Evolution) (Volume 1).* Language Science Press.

[28] L. Steels and T. Belpaeme. 2005. Coordinating Perceptually Grounded Categories through Language: a Case Study for Colour. *Behavioral and Brain Sciences* 28 (2005), 469–529.

[29] L. Steels and E. Garcia-Casademont. 2015. How to play the syntax game. In *Proceedings of the European Conference on Artificial Life.* MIT Press, 479–486.

[30] L. Wittgenstein. 1953. *Philosophical Investigations.* Macmillan, New York.