Incremental simulations of the emergence of grammar: towards complex sentence-meaning mappings

Andrei Popescu-Belis and John Batali

University of California at San Diego
Department of Cognitive Science
9500 Gilman Drive, La Jolla, CA 92093-0515
{apopescu, batali}@cogsci.ucsd.edu

Introduction

Experiments with societies of communicating agents have shown that various communication conventions can emerge in order to express the structure of situations in an environment (e.g., Batali 1998, Steels 1997). However, it is often unclear how much implicit knowledge is initially given to the agents, or may come from the way meaning itself is encoded. In this study, we analyze two experimental models from the point of view of built-in knowledge and emergent capacities. The first one proves the emergence of syntax-specific capacities in agents that initially possess only semantic knowledge; the second model incorporates part of these capacities as initial knowledge. This gives its agents similar capacities in a shorter time, thus opening the way to more complex semantic structures, in the conceptual graphs formalism.

We defend an incremental paradigm for building models: to simulate increasingly expressive communication codes, it is possible to avoid evolving them from scratch, and start at a level reached by previous experiments. If a certain level of complexity has been attained, then it may not be useful to go again through all levels in a further simulation, but only through the new ones. This paradigm provides gain in simulation time that may prove crucial, and a better control over the intermediate states.

Semantic representations

The experiments described here presuppose that the agents are able to represent situations in a conceptual form, their purpose being an agreement on mappings between these forms and messages (sequences of letters). One of the most basic features of language, though not intrinsic to communication codes, is the division of this mapping procedure in several stages. Thus, messages are segmented into lexical units; most of these have a proper meaning (lexical semantics); and their composition yields a complex, non-additive meaning (propositional semantics). A computational simulation of the emergence of language has to account for these levels – and possibly also for other related phenomena (discourse, pragmatics, conceptual system, etc).

The first experiment (Batali, *in press*) uses formula sets, i.e. conjunctions of <feature, referent(s)> formulae. The referents are numbers designating the participants in the situation. The unary features represent characteristics of a referent, while the binary features represent relations between two referents. For instance, the formula set *{(goose 1) (sang 1) (noticed 1 2) (snake 2)}* can be glossed as “A goose that sang noticed a snake”. This formalism is equivalent to a small subset of the first order predicate logic. It also provides a straightforward representation of the referents and

could be linked to an agent's perceptual device (cf. experiment of Steels and Kaplan, *in press*). However, it has to be extended to represent more complex semantic aspects, as well as referring status (the previous example could also be glossed as "*The* goose that sang noticed *the* snake").

In an attempt to use a better known and more expandable representation, the second experiment uses conceptual graphs (CG), i.e. concept nodes and relations between them (Sowa 1992). Three relation types match the argument slots of the formulae. Hence, the previous situation is represented as:

[NOTICED] → (AGENT) → [GOOSE: #1] → (ATTRIBUTE) → [SINGING]
→ (PATIENT) → [SNAKE: #2]

For now, we use simplified CGs without the referent numbers: [GOOSE] stands for [GOOSE: #1], "the goose". These CGs are equivalent to formula sets. The full CGs, to be used later, can represent the referring status using notations in the concept nodes (for "a snake", "the snake", "some snakes", "three snakes", "all snakes").

Emergence of fundamental syntactic properties

In the first experiment, the agents are given situations (formula sets), and for each dialog the sender, chosen at random, has to produce a sequence of letters representing the situation. Therefore, it either creates a new *exemplar* (meaning-to-string mapping) or uses combination and/or substitution on the existing ones. If the receiver is in learning mode, it uses both the situation and the message to update its own exemplar set. In trial mode, it has to infer the situation described by the received message. If it succeeds, the overall communicative accuracy or success increases.

The built-in *parsing/generation mechanism* makes an agent capable of replacing part of an exemplar with another exemplar (substitution). Thus, strings and situations get gradually broken up in their irreducible constituents. Conversely, exemplars are also put together to fit new situations (combination). However, without a built-in *learning device*, an agent would only store unstructured mappings from meanings to strings. This retribution device makes the creation of such exemplars more costly than the reuse of elementary exemplars through substitutions and combinations. Only frequently used exemplars are reinforced, provided they lead to a correct interpretation.

Incremental design: use of the emergent grammars as starting points

The first experiment starts with a "complexive" use of language: holistic mappings from strings to situations, without intermediate levels. Its results prove that in each population, a set of exemplars acting as grammatical rules emerges, each one corresponding to a situation model. In general, their internal structure is partitioned: the formulae are grouped according to the referents. Conversely, there is also an agreement on a lexicon, or strings associated to singleton formulae, which are substituted on the complex exemplars. Sometimes, strings playing only grammatical roles appear.

Any of the states emerging in low-knowledge conditions may be used at a starting point for new experiments. They authorize us to consider separately the study of lexical and of propositional semantics. Furthermore, word segmentation and word understanding can also be separated (figure 1). This built-in knowledge allows for faster convergence, easier implementation, and use of more complex semantic descriptions.

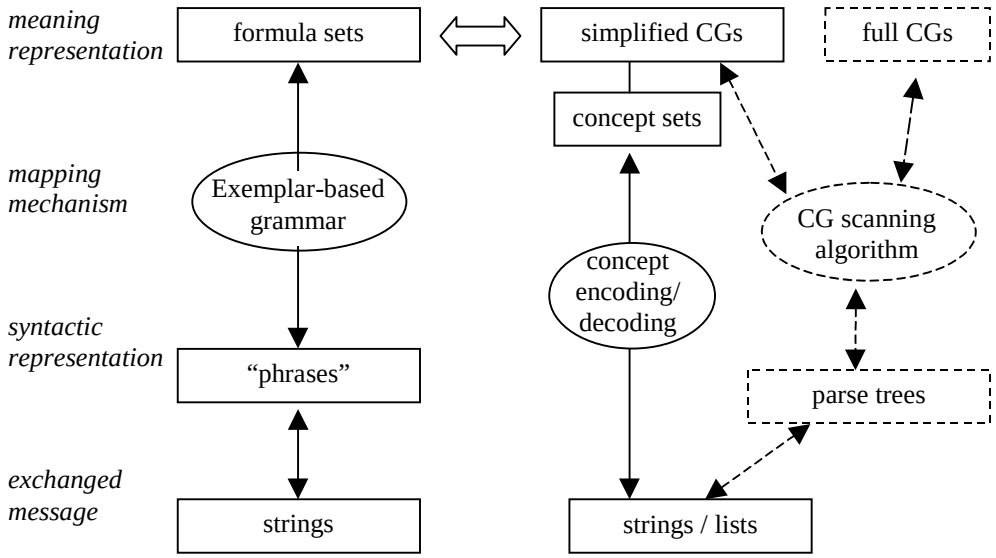

***Figure 1**. Data structures and conversion mechanisms in the two experiments.*

Emergence of word-concept associations

In a first series of trials, we focused on lexical semantics conventions: the agents had to map a message to an unstructured set of concepts constituting the situation. General agreement ("convergence") on words has been observed in a variety of situations, using the same dialog protocol as in the first experiment. An agent's *conservatism* (in [0, 1]) is one of the main parameters governing convergence time. Other parameters include the number of words an agent is allowed to guess in a learning dialog (at least one), and the size of the situations that the agents observe (e.g., random size between one and five concepts). Their effects are undergoing extensive theoretical and experimental study. Here, figure 2 shows the difference in average convergence time between populations that use pre-segmented vs. concatenated messages. Both options converge, but the first is faster, and the difference increases with population size.

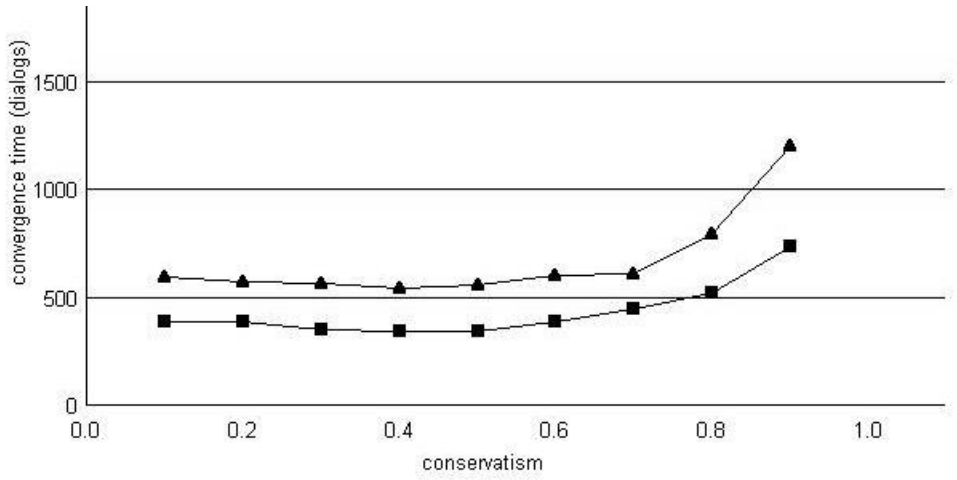

***Figure 2**. Average convergence times depend on the agents' lexical conservatism. The lower curve is for separate words, the upper for concatenated ones (15 concepts, 5 agents, 1 to 4 concept situations, agents allowed to guess 1 or 2 words).*

Towards the emergence of syntactic conventions

Once words are mapped to concepts, the structure of situations is represented using a lexicalized tree grammar adapted to the simplified CGs (Popescu-Belis 1999, Allexandre and Popescu-Belis 1998). Each concept has an associated elementary tree – branch order being a syntactic parameter – and these are combined using substitutions and adjunctions. Despite a huge number of parameter combinations, this TAG-like grammar allows for much less variation than the exemplar-based one.

To understand a message, the agents use a two-phase inference mechanism. The receiver first maps the words to the situation's concepts, then performs a trial and error comparison between the message it could generate for these concepts. These modules are under implementation, but based on previous experiments, we expect an agreement on branch orders. Of particular interest are the measures of the convergence rate as a function of the number of concepts, and the comparison with the first experiment.

Conclusion

The grammars of the two experiments both fulfill analog tasks in individual agents (figure 1). However, while the first experiment shows how grammatical conventions emerge in low-knowledge populations, the second makes use of this experimental proof to seek faster convergence and a more open conceptual formalism. An exemplar-based grammar for conceptual graphs is possible, but would be long to converge, and its final shape could not be well controlled. On the contrary, faster convergence and a more constrained form allow us to consider environments that are more complex.

References

Allexandre C. & Popescu-Belis A. (1998) – "Emergence of Grammatical Conventions in an Agent Population Using a Simplified Tree Adjoining Grammar", *Proceedings of ICMAS'98*, Paris, p.383-384.

Batali J. (1998) – "Computational Simulations of the Emergence of Grammar", in Hurford J.R., Studdert-Kennedy M. & Knight C. (eds.), *Approaches to the Evolution of Language*, Cambridge, p.405-426.

Batali J. (*in press*) – "The Negotiation and Acquisition of Recursive Grammars as a Result of Competition Among Exemplars", in Briscoe T. (ed.), *Linguistic Evolution Through Language Acquisition: Formal and Computational Models*, Cambridge.

Popescu-Belis A. (1999) – *Modélisation multi-agent des échanges langagiers : application au problème de la référence et son évaluation*, Ph.D. thesis, Université de Paris XI.

Sowa J. (1992) – "Conceptual Graphs Summary", in *Conceptual Structures: Current Research and Practice*, Melksham.

Steels L. (1997) – "The Origins of Syntax in Visually Grounded Robotic Agents", *Proceedings of IJCAI'97*, Nagoya, Japan, vol. 2/2, p.1632-1641.

Steels L. & Kaplan F. (in press) – "Bootstrapping Grounded Word Semantics", in Briscoe T. (ed.), *Linguistic Evolution Through Language Acquisition: Formal and Computational Models*, Cambridge.