Chapter XIII
Propositional Logic Syntax Acquisition Using Induction and Self-Organisation

Josefina Sierra
Universidad Politécnica de Cataluña, Spain

Josefina Santibáñez
Universidad de La Rioja, Spain

ABSTRACT

This chapter addresses the problem of the acquisition of the syntax of propositional logic. An approach based on general purpose cognitive capacities such as invention, adoption, parsing, generation, and induction is proposed. Self-organisation principles are used to show how a shared set of preferred lexical entries and grammatical constructions, that is, a language, can emerge in a population of autonomous agents which do not have any initial linguistic knowledge. Experiments in which a population of autonomous agents constructs a grammar that allows communicating the formulas of a propositional logic language are presented. These experiments extend previous work by considering a larger population and a much larger search space of grammar rules. In particular, the agents are allowed to order the expressions associated with the constituents of a logical formula in arbitrary order. Previous work assumed that the expressions associated with the connectives should be placed in the first position of the sentence.

INTRODUCTION

Recent work in linguistics and artificial intelligence (Steels, 1998, 2000, 2004; Batali, 2002; Kirby 2002) has suggested that some of the complex structure of language may be the result of a quite different process from biological evolution. Interesting experiments showing the emergence of compositional and recursive syntax in populations of agents without initial linguistic knowledge have been presented as evidence in support of alternative explanations. This chapter combines general purpose cognitive capacities (e.g., invention, adoption, parsing, generation and induction) and self-organisation principles proposed as effective mechanisms for syntax acquisition in these experiments in order to address the problem of the acquisition of the syntax of propositional logic.

The important role of logic in knowledge representation and reasoning (McCarthy, 1990) is well known in artificial intelligence. Much of the knowledge used by artificial intelligent agents today is represented in logic, and linguists use it as well for representing the meanings of words and sentences. This chapter differs from previous approaches in using the syntax of logic as the subject of learning. Some could argue that it is not necessary to learn such a syntax, because it is built in the internal knowledge representation formalism used by the agents. We'd argue on the contrary that logical connectives and logical constructions are a fundamental part of natural language, and that it is necessary to understand how an agent can both conceptualise and communicate them to other agents.

The research presented in this chapter assumes previous work on the **conceptualisation of logical connectives** (Piaget, 1985; Santibáñez, 1984, 1988, 1989). In (Sierra 2001, 2002) a grounded approach to the acquisition of logical categories (connectives) based on the **discrimination** of a "subset of objects" from the rest of the objects in a given context is described. The "subset of objects" is characterized by a logical formula constructed from perceptually grounded categories. This formula is satisfied by the objects in the subset and not satisfied by the rest of the objects in the context. In this chapter we only focus on the problem of the acquisition of the syntax of propositional logic, because it is a necessary step to solve the complete problem of the acquisition of a grounded logical language (encompassing the acquisition of both the syntax and the semantics of propositional logic).

The rest of the chapter is organised as follows. First we present the formalism used for representing the grammars constructed by the agents. Then we describe in some detail the language games played by the agents, focusing on the main cognitive processes they use for constructing a shared lexicon and grammar: invention, adoption, induction and self-organisation. Next we report the results of some experiments in which a population of autonomous agents constructs a **shared language** that allows communicating the formulas of a propositional logic language. Finally we summarize some related work and the main contributions of the chapter.

GRAMMATICAL FORMALISM

We use a restricted form of Definite Clause Grammar in which non-terminals have three arguments attached to them. The first argument conveys semantic information. The second is a score in the interval [0, 1] that estimates the usefulness of that association in previous communication. The third argument is a counter that records the number of times the association has been used in previous language games.

Many grammars can be used to express the same meaning. The following holistic grammar can be used to express the propositional formula right∧light.

s([and, right, light]), 0.01) → andrightlight (1)

This grammar consists of a single rule which states that 'andrightlight' is a valid sentence meaning right∧light. Notice that we use Prolog grammar rules for describing the grammars. The semantic argument of non-terminals uses Lisp like notation for representing **propositional formulas** (e.g., the Prolog list [and, [not, right], light] is equivalent to ¬right∧light). The third argument (the use counter) of non-terminals is not shown in the examples.

The same formula can be expressed as well using the following compositional, recursive grammar: s is the start symbol, c1 and c2 are the names of two syntactic categories associated with unary and binary connectives, respectively. Like in Prolog, variables start with a capital letter and constants with a lower case letter.

s(light, 0.70) → light (2)
s(right, 0.25) → right (3)
s(up, 0.60) → up (4)
c1(not, 0.80) → not (5)
s([P, Q],S) → c1(P, S1), s(Q, S2),
{S is S1∗S2∗0.10} (6)
c2(or, 0.30) → or (7)
c2(and, 0.50) → and (8)
c2(if, 0.90) → if (9)
c2(iff, 0.60) → iff (10)

Related Content