A Constructive Mathematics approach for Natural Language formal grammars

Federico Gobbo and Marco Benini
{federico.gobbo,marco.benini}@uninsubria.it
Department of Computer and Communication Sciences
University of Insubria
Varese, Italy

Short abstract

A mathematical description of natural language grammars has been proposed first by Leibniz. After the definition given by Frege of unsaturated expression and the foundation of a logical grammar by Husserl, the application of logic to treat natural language grammars in a computational way raised the interest of linguists, for example applying Lambek's categorial calculus. In recent years, the most consolidated formal grammars (e.g., Minimalism, HPSG, TAG, CCG, Dependency Grammars) began to show an interest in giving a strong psychological interpretation to the formalism and hence to natural language data on which they are applied. Nevertheless, no one seems to have paid much attention to cognitive linguistics, a branch of linguistics that actively uses concepts and results from cognitive sciences. Apparently unrelated, the study of computational formalisms has developed in pair with constructive formal system, especially in the branch of logic called proof theory, see, e.g., the Curry-Howard isomorphism and the typed functional languages. In this paper, we want to bridge these worlds and thus we present our natural grammar formalism, called Adpositional Grammars (AdGrams), that is founded over both cognitive linguistics and constructive mathematics.

First choice for track: IX. Logic and Computation.
Second choice for track: VI. Crossroads.

Extended abstract

A mathematical description framework of natural language grammars has been proposed first by Leibniz, who tried to catch the laws of human thought through his *characteristica universalis* and an (almost) ambiguity-free planning of Latin called *lingua generalis*, to be used as a written medium for European scholars of its time. After the definition given by Frege of unsaturated expressions, Husserl gave the foundation of a logical grammar based on its formal constituents, the meaning categories. Along with the development of Computer Science, linguistics started to be interested in formalizing grammars through mathematics, for example basing themselves on Lambek's categorial calculus (Morrill 2007).

Therefore, different approaches based on mathematics and logics have consolidated their own methodology and communities of practice: at least, Minimalism, Tree-Adjoining Grammar (TAG), Head-Driven Phrase Structure Grammar (HPSG), various Categorial grammar frameworks (TLG, TTG, CCG and Pregroup), Dependency Grammar. These approaches are somewhat similar, someway different. In particular, in recent years they started to show an interest in giving a strong form of psychological interpretation to the formalism and hence to natural language data on which they are applied. On the other hand, although cognitive linguistics is an established branch of linguistics from more than 20 years (Croft-Cruse 2004), whose aim is to apply well-known concepts from psychology and cognitive sciences to linguistics, there is no explicit link between the community of cognitive linguists and the communities of formal grammarians.

In recent days, we have proposed a novel grammar formalism called Adpositional Grammars (AdGrams), which aims to fill this gap between cognitive linguistics and formalization of natural language grammars (Gobbo 2009).

On a linguistic point of view, AdGrams move from the early days of the Dependency Grammar tradition, i.e., the classic by Tesnière (1959), but it refines the concept of 'dependency' through the dichotomy trajector/landmark by Langacker (1987), borrowed by the German school of Gestalt psychology, e.g. Kurt Koffka or Max Wertheimer. A *trajector* is the most salient participant put in the focused position, while its reference point of observation is called the *landmark*. Evidently, a trajector cannot exist without a landmark. A trajector can be applied on a landmark (*application*) or, on the contrary, a landmark can exist in advance in the mental space of the speaker and the trajector emerges (*retroapplication*). What we succeeded to configure is a Porphyrian (strictly binary) tree structure where application and retroapplication are the two basic ways to relate trajectors and landmarks. Specifically, in all natural languages there are some grammatical morphemes which are specialized into convey the appropriate trajector/landmark relation. Typically, they are prepositions in Hindo-European languages, postpositions in languages such as Japanese or Turkish, according to Pennacchietti (2008), who first explored this research line in linguistics, especially for Semitic languages. We have refined and generalized his model so to comprehend not only prepositions and postpositions but also other grammatical morphemes such as conjunctions – in brief, *adpositions*. In fact, AdGrams are able to represent the cognitive relation between trajectors and landmarks both at a phrasal and syntagmatic level. Let us explain it through an example.

1. Alice is rich, **so** she can pay.
2. Alice can pay, **because** she is rich.

The two sentences convey the same meaning but through a different construal, i.e., a different cognitive way to underline the relevant information (Croft-Cruse 2004). In fact, the relevant information is that Alice can pay (trajector) while the fact that she is rich is less relevant (landmark), but the two adpositions *so* and *because* configure the relation in different manners, and in fact this is expressed with a different word order.

From a computational point of view, it seems reasonable to use techniques of constructive mathematics in order to build a formal grammar, since constructive logics are the natural framework to model computation, as argumented in Troelstra (1988) and in Barendregt (1992). This idea is not a complete novelty, as traced in Morrill (2007). In particular, defining a grammar in the language of logic as a specification and letting the semantics of logic to act as a computational engine, which is possible due to the constructive nature of the logical system, allows to parse/generate/manipulate the sentences essentially for free. This phenomenon, where the (proof-theoretic) semantics of a constructive logic allows the interpretation of formulae as computable specification, has been successfully used in the field of Formal Methods to develop modern forms of declarative programming. In this respect, we have written an instance of AdGrams in an appropriate logical formalism, based on intuitionistic logic, together with a specialized semantics. This instance, a first test, is fit for the Quasi-Natural Language (QNL, Lyons 2006) Esperanto, and it proved to cover approximately 95% of the available corpora of language-in-use, except of the well-known open issues of quoting and name-entity recognition (Gobbo 2009). Finally, it is very likely that our instance computes in polynomial space, and so it can be compared with other formal grammars, as TAG or CCG.

In the full version of this paper we will explain why this line of research is interesting, the details on the foundation of the formal system, in order to show how we have put together logics, constructive mathematics and cognitive linguistics to obtain a parser based on AdGrams.

Essential bibliography

Barendregt, Henk (1992). *Lambda calculi with types*. In *Handbook of Logic in Computer Science vol 2*, Abramsky, S., Gabbay, Dov M., Maibaum, T.S.E. editors. Oxford University Press.
Croft, William and Cruse, D. Alan (2004). *Cognitive Linguistics*. Cambridge University Press.
Gobbo, Federico (2009). *Adpositional Grammars*. PhD dissertation. University of Insubria.
Langacker, Ronald W. (1987). *Foundations of Cognitive Grammar, volume 1. Theoretical Prerequisites*. Stanford University Press.
Lyons, John (2006). *Natural language and universal grammar*. Cambridge University Press.
Morrill, Glyn (2007). A Chronicle of Type Logical Grammar: 1935--1994, *Research on Language and Computation*, 5, 3:359–386.
Pennacchietti, Fabrizio A. (2008). Preposizione semitiche tra diacronia e sincronia. *Aula Orientalis*, XXVI(1):143–160.
Tesnière, Lucien (1959). *Eleménts de syntaxe structurale*. Editions Klincksieck, Paris.
Troelstra, Anne Sjerp (1988). *Constructivism in Mahtematics - vol I and II*. Elsevier.