

# Semantic-Driven Emergence of Syntax : The Principle of Compositionality upside-down

Isabelle Tellier

LIFL and Université Charles de Gaulle-lille3 (UFR IDIST)

59 653 Villeneuve d'Ascq Cedex, FRANCE

tel : 03-20-41-61-78

fax : 03-20-41-61-71

E-mail : [tellier@univ-lille3.fr](mailto:tellier@univ-lille3.fr)

<http://www.grappa.univ-lille3.fr/~tellier/>

## .Introduction

In Chomskyan views of linguistics, syntax is a preeminent independent level whose knowledge is mainly innate. Semantics is then usually seen as a secondary structure deduced from the syntactic analysis. The Principle of Compositionality is, in this context, a precise way of specifying the passage from syntax to semantics.

But from an emergence perspective, this conception is very awkward. As a matter of fact, the ability to communicate meanings is of far higher priority than the ability to build a formal grammar, so *semantics must precede syntax*.

We still assume a computational point of view on syntax and semantics, but we propose to reinterpret the Principle of Compositionality to show that *syntax can derive from semantics*. The idea, inspired by recent results in the domain of Machine Learning, is to prove that a grammar can be completely specified by the description of the way semantic items (roughly corresponding with word meanings) are combined into global meanings. We thus provide theoretical arguments to avoid strong hypotheses about the innateness of syntax.

## .The Principle of Compositionality

### . Intuitive formulation

The Principle of Compositionality, mainly known by linguists and logicians, allows to characterize the connection between the syntax and semantics of natural languages. It is usually (and seemingly wrongly) attributed to Frege ([Janssen 97]). Its contemporary version states that : "the meaning of a compound expression is a function of the meaning of its parts and of the syntactic rules by which they are combined" ([Partee 90]). It has been the basis of several formal theories in computational linguistics, among which the best known may be Montague's semantics ([Montague 74], [Dowty 81]).

If the "parts" mentioned in the definition are assimilated with morphemes (or, to simplify, with words), and the "compound expressions" with phrases (which is the usual interpretation), this formulation implies that :

- words have individual meanings ;
- the meaning of a phrase (and thus of a sentence) *only* depends of the meaning of its words and of its syntactic structure.

The Principle of Compositionality has strong psychological justifications, as it "can explain how a human being can *understand* sentences never heard before" ([Janssen 97]).

**. Formal definition**

The previous definition can be specified in a more formal way, inspired by [Montague 74] and [Janssen 97]. Two mappings need to be defined :

- a mapping associating each *word* with a *meaning*, i.e. a *semantic item* ;
- a mapping associating each *syntactic rule* (or, more generally, each *class* of syntactic rules) with a *semantic composition* (i.e. a function applying on semantic expressions and producing other semantic expressions).

Figure 1 shows on an abstract example how these mappings are combined to define a structure-preserving correspondence between syntactic and semantic trees. On the left tree, indexes  $g_1$  and  $g_2$  denote two classes of syntactic rules and on the right one,  $h_1$  and  $h_2$  denote the corresponding semantic compositions. If the grammar and both mappings are known, then the global meaning  $h_2(h_1(\text{meaning}_1, \text{meaning}_2), \text{meaning}_3)$  of the sentence "word<sub>1</sub> word<sub>2</sub> word<sub>3</sub>" can be automatically computed.

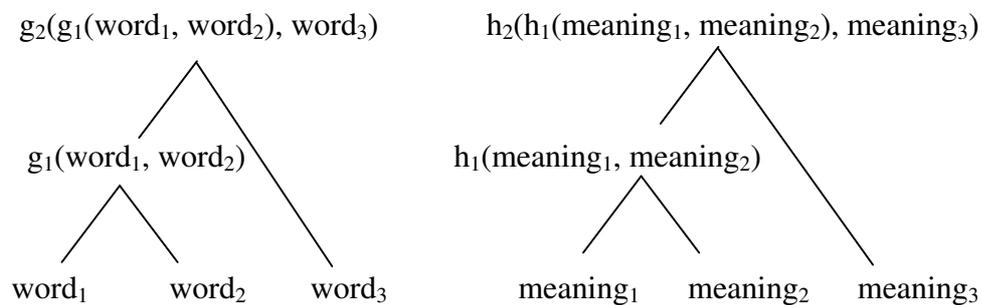


Figure 1 : application of the Principle of Compositionality on an abstract example

In cases of syntactic ambiguities, each different syntactic tree is associated with a different semantic tree whose global meaning may be different. Lexical ambiguities, due to polysemic words, can be handled by considering as many *copies* of the polysemic words as they have different meanings. Both mappings are in this case *bijective*.

Although this classical formal version of the Principle of Compositionality is only used as a one-sided way (*from syntax to semantics*), note that it is stated as a *tree isomorphism*. Thus, nothing prevents us from going *against* the usual stream.

**.Turning upside-down the Principle of Compositionality**

**. Theoretical statements**

We believe that the emergence of natural languages was mainly motivated by the need to convey not only atomic invariable meanings but also complex functional combinations of these semantic items. We will now show that syntactic structures can be considered as a *direct consequence* of such a need to combine meanings.

For this, we first have to turn upside-down the Principle of Compositionality. This means that what is now supposed to be given is a set of semantic items and a set of semantic compositions noted  $\{h_i\}_{1 \leq i \leq n}$ . We then consider two new mappings :

- a mapping associating each *semantic item* with a *symbol*, which can also be called, in Saussure' s terms, *signifier* ;
- a mapping associating each *semantic composition*  $h_i$  with a *class index* noted  $g_i$ , for each  $1 \leq i \leq n$ .

These mappings are again bijective : they are the reversed versions of the previous two mappings. The result of applying this reversed version of the Principle of

Compositionality on a semantic combination is a tree structure whose leaves are symbols and whose nodes are indexed by the members of a finite set of elements. This resulting tree is *not*, in most cases, a full syntactic tree, but a simplified version of it, where usual non terminal symbols are replaced by class indexes. The interesting point is that this structure exactly coincide with a recently emerged notion in the domain of Grammatical Inference : the notion of *Structural Example*.

In the domain of Grammatical Inference, subpart of Machine Learning, the purpose is to identify a formal grammar from sentences it generates. Various theoretical results tend to prove that strings of words are not informative enough to specify a unique formal grammar ([Gold 67], [Valiant 84]). But Structural Examples, i.e. parenthesized strings of words with eventual class indexes, allow to achieve this goal : there exist algorithms able to converge towards the description of the unique formal grammar compatible with a set of Structural Examples ([Sakakibara 90 & 92], [Kanazawa 96]).

• **A detailed example**

Our new version of the Principle of Compositionality applies *from semantics to syntax*. For sake of simplicity, let meaning representations be expressed by logical formulas. We note *John'* and *Mary'* two logical individual constants and *run<sub>1</sub>'* and *love<sub>2</sub>'* two logical predicates of arity respectively 1 and 2. By convention, we suppose that the first argument of a two-place predicate coincide with its direct object and the second one with its grammatical subject. We suppose that the admitted semantic compositions  $h_1$  and  $h_2$  are *oriented functional applications* defined as follows :

- for any couple of semantic expressions a and b,  $h_1(a, b)=a(b)$  ;
- for any couple of semantic expressions a and b,  $h_2(a, b)=b(a)$ .

The logical proposition :  $run_1'(John')$ , denoting the fact that "John runs", can be obtained from the semantic items and from the semantic compositions  $h_1$  and  $h_2$  in two ways :  $run_1'(John') = h_1(run_1', John') = h_2(John', run_1')$ . By the upside-down Principle of Compositionality, those two ways are respectively associated with two Structural Examples :  $g_1(run, John)$  and  $g_2(John, run)$ , where "run" and "John" are the signifiers respectively associated with the semantic items  $run_1'$  and  $John'$ . Of course, the first structure will give rise to a grammar where verbs precede their grammatical subject and the second one to a grammar where grammatical subjects are uttered first.

Similarly,  $love_2'(Mary')(John')$ , expressing the fact that "John loves Mary" can be obtained in six various ways, each one corresponding with a possible ordering of a subject S, a verb V and a direct object O :

- $love_2'(Mary')(John') = h_2(John', h_1(love_2', Mary'))$  : SVO
- $love_2'(Mary')(John') = h_2(John', h_2(Mary', love_2'))$  : SOV
- $love_2'(Mary')(John') = h_1(h_2(Mary', love_2'), John')$  : OVS
- $love_2'(Mary')(John') = h_1(h_1(love_2', Mary'), John')$  : VOS

For the last two possible orderings, because of our notational convention, we need the semantic item  $\lambda x \lambda y.love_2'(y)(x)$  where lambda abstractions allow to invert the order of the arguments of the predicate, instead of  $love_2'$  :

- $love_2'(Mary')(John') = h_2(Mary', h_2(John', \lambda x \lambda y.love_2'(y)(x)))$  : OSV
- $love_2'(Mary')(John') = h_1(h_1(\lambda x \lambda y.love_2'(y)(x)John'), Mary')$  : VSO

The last two constructions are less frequently found in natural languages than the others. Each of these compositions specifies a unique Structural Example.

Combining semantic items in regular ways means that, if the semantic composition chosen to express "John runs" is  $h_1(run_1', John')$ , then the semantic composition chosen to express "John loves Mary" should be one where the predicate also precedes its first

argument, and where the semantic composition  $h_1$  is used to combine both items.

It has been proved in [Kanazawa 98] that huge sub-classes of context-free grammars are identifiable (in the sense of [Gold 67]) from Structural Examples built on the model of this example (i.e. based on a two-classes partition of the set of syntactic rules).

## .Conclusion

This work suggests a scenario for the emergence of syntax. The first step is the association of *symbols* (or signifiers) with semantic items. Computational simulations of this process have already been proposed ([Siskind 97]). The second step is the intention to communicate *combined meanings* built from the semantic items. The language of semantic representation is then supposed to be first acquired (or innate), but its syntax is much simpler than the one of natural languages. If these combined meanings are obtained in regular ways, then the definition of these combinations is equivalent, using our reversed version of the Principle of Compositionality, with the specification of a set of Structural Examples. In a last step, this sample of Structural Examples naturally leads to the description of a unique formal grammar. The only innate structures supposed are the semantic compositions : in our example, two very general functional applications are enough to explain the various possible orderings of phrases in natural languages.

Each natural language then appears as the result of choices made *at the semantic level* and reflected at the syntactic one through the upside-down Principle of Compositionality. Of course, other parameters than the order of phrases should be considered to distinguish one language from another one, and many other features could not be detailed here, but the upside-down Principle of Compositionality seems an ideal underlying mechanism allowing to connect semantic combinations with syntactic structures.

## .References

- [Dowty 81] : D. R. Dowty, R. E. Wall, S. Peters, *Introduction to Montague Semantics*, Reidel, Dordrecht, 1989.
- [Gold 67] : M. Gold, "Language Identification in the Limit", *Information and Control* 10, P447-474, 1967.
- [Janssen 97] : T. M. V. Janssen, "Compositionality", in : *Handbook of Logic and Language*, Elsevier, Amsterdam and MIT Press, Cambridge, J. Van Benthem & A. ter Meulen (Eds), p417-473, 1997.
- [Kanazawa 98] : M. Kanazawa, "Learnable classes of Categorical Grammars", CSLI Publications and FoLLI, *Studies in Logic, Language and Information*, 1998.
- [Montague 74] : R. Montague, *Formal Philosophy; Selected papers of Richard Montague*, Yale University Press, New Haven, 1974.
- [Partee 90] : B. Partee, A. ter Meulen, R. E. Wall, *Mathematical methods in Linguistics*, in "Studies in Linguistics and Philosophy" n°30, Kluwer, Dordrecht, 1990.
- [Sakakibara 90] : Y. Sakakibara, "Learning context-free grammars from structural data in polynomial time", *Theoretical Computer Science* 76, p223-242, 1990.
- [Sakakibara 92] : Y. Sakakibara, "Efficient Learning of Context-Free Grammars from Positive Structural Examples", *Information and Computation* 97, p23-60, 1992.
- [Siskind 97] : J. M. Siskind : "Learning Word Meanings", in *Computational Approaches to Language Acquisition*, M. Brent (Ed), MIT Press, 1997.
- [Valiant 84] : L. G. Valiant, "A theory of the learnable", *Communication of the ACM*, p1134-1142, 1984.