In A. Gelbukh (ed.), *Computational Linguistics*, 2001, Springer, 18-31.

A Fully Lexicalized Grammar for French based on Meaning-Text Theory

Sylvain Kahane

CNRS & Lattice-Talana, Université Paris 7, UFRL, case 7003,
2, place Jussieu, 75251 Paris Cedex 05, France
sk@ccr.jussieu.fr
http://www.linguist.jussieu.fr/~skahane

Abstract. The paper presents a formal lexicalized dependency grammar based on Meaning-Text theory. This grammar associates semantic graphs with sentences. We propose a fragment of a grammar for French, including the description of extractions. The main particularity of our grammar is it that it builds bubble trees as syntactic representations, that is, trees whose nodes can be filled by bubbles, which can contain others nodes. Our grammar needs more complex operations of combination of elementary structures than other lexicalized grammars, such as TAG or CG, but avoids the multiplication of elementary structures and provides linguistically well-motivated treatments.[1]

1 Introduction

Meaning-Text theory (MTT) has been developed since more than thirty years, but no complete formalization of the model has been achieved. Our main goal in this paper is to propose a formal grammar based on MTT. We insist on the fact that our grammar in any case is a 'kosher' implementation of MTT.

Following the MTT postulates ([16]: 53), we consider that 1) a grammar[2] is a formal system which ensures the bidirectional correspondence between texts and meanings (= semantic representations) and that 2) intermediate levels of representation—a morphological level and a syntactic level—must be considered and that a grammar consists of several modules which establish correspondence between representations of adjacent levels. Our grammar is composed of three modules: the morphological module ensures the correspondence between sentences and morphological representations, the syntactic module ensures the correspondence between morphological and syntactic representations and the semantic module ensures the correspondence between syntactic and semantic representations.

In section 2, we present the different levels of representation, in section 3, the syntactic module and in section 4, the semantic module. Our formalism will be exem-

[1] I want to thank Jasmina Mili evi , as well as two anonymous referees, for many valuable comments and corrections.

[2] The term grammar is used in its Chomskian sense of 'linguistic model'.

plified by a fragment of French grammar. Extractions, which need extensions of the formalism, will be treated separately in section 5.

2 Different Levels of Representation of a Sentence

A *morphological representation* of a sentence is the sequence of the morphological representations of the words of the sentence; the morphological representation of a word is *surface lexical unit* accompanied with a list of *surface grammemes*. Consider the sentence:

(1) *Zoé a parlé à un type étrange.*
Zoé has talked to a guy strange
'Zoé has talked to a strange guy'

The morphological representation of (1) is:

(2) *ZOÉ*$_{sg}$ *AVOIR*$_{ind,présent,3,sg}$ *PARLER*$_{part\_passé}$ *À UN*$_{masc,sg}$ *TYPE*$_{sg}$ *ÉTRANGE*$_{masc,sg}$

The *syntactic representation* of a sentence is a non ordered *dependency tree* similar to the surface syntactic trees of MTT ([16]) or the stemmas of Tesnière ([23]). The nodes of the structure are labeled with surface lexical units, each being accompanied with a list of *surface grammemes,* and the dependencies are labeled with (surface) *syntactic relations*.

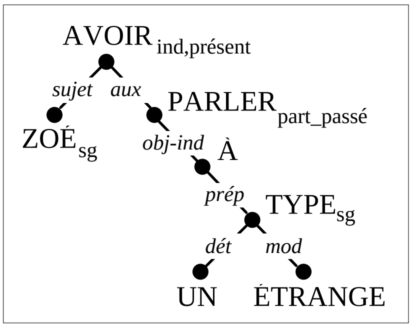


Fig. 1. *Syntactic representation of (1)*

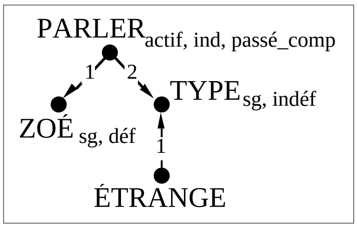


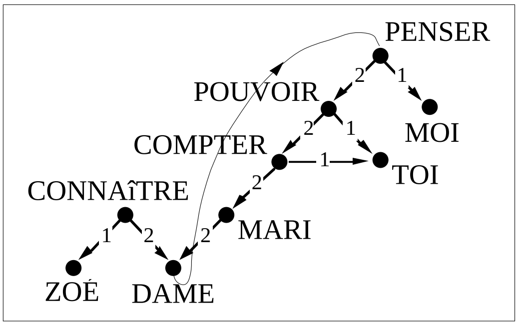
Fig. 2. *The semantic representation of (1)*

The *semantic representation* of a sentence is a directed graph whose nodes are labeled by deep lexical units, each being accompanied by a list of deep grammemes. A *deep lexical unit* corresponds to a surface lexical unit or a group of surface lexical units making an idiom. A *deep grammeme* is a grammeme with a meaning (including voice grammemes, which do not exactly express a meaning, but depend on semantic communicative choices). A deep grammeme can correspond to a surface grammeme or a complex expression including surface lexical units: that is the case of the French *passé composé* (AVOIR$_{ind,présent}$ + V$_{art_passé}$) or the French passive voice (ÊTRE + V$_{part_passé}$). A deep lexical unit acts like a predicate and is linked to its arguments by arrows pointing on them. The different arrows emerging from a deep lexical unit are numbered from 1 to *n* following the increasing syntactic salience of the arguments.

Such an arrow, representing a predicate-argument relation, is called a *semantic dependency*; the predicate is *the semantic governor* and its argument, the *semantic dependent*. Our semantic representation is a compromise between the semantic and deep syntactic representations of MTT. On the one hand, the nodes of our semantic representation are labeled with deep lexical units, rather than by semantemes (= the meaning of a deep lexical unit)[3]. On the other hand, a hierarchy on the nodes is superimposed to the graph structure; this hierarchy, which expresses the communicative importance of the meanings expressed in the sentence, is called the *communicative dependency*, following [21][4]. In our figures, the communicative hierarchy is expressed by placing a communicative governor above its communicative dependents. Therefore, a downward arrow indicates semantic and communicative dependencies in the same direction, the (communicative and semantic) dependent being an *actant* (or argument), while a upward arrow indicates semantic and communicative dependencies in opposite directions, the communicative dependent (and semantic governor) being a *modifier*. With this hierarchy, our semantic representation is close from a TAG derivation tree (cf. [2] for a comparison between TAG and MTT).

In the previous example, every communicative dependency is superimposed with a semantic dependency, which allows us to indicate the communicative dependencies with the upward or downward direction of the semantic dependencies. Relative clauses are more complex. Consider:

(3) *Zoé connaît la dame sur le mari de laquelle je pense que tu peux compter.*
Zoé knows the woman on the husband of whom I think you can count
'Zoé knows the woman whose husband I think you can count on'



[3] The initial objective of the MTT semantic representation is to be a semantic invariant of a set of paraphrases, which requires attributing two synonymous lexical units to the same semanteme. We prefer considering that there is one-to-one correspondence between deep lexical units and semantemes and the replacement of a lexical unit by a synonymous lexical unit is a paraphrase rule (see Mel'čuk 1992 for a presentation of these rules).

[4] In the MTT tradition, only the communicative top nodes of some sub-graphs (such as the rheme and the theme) are indicated. The communicative top node (= the dominant node of Polguère 1990) of a phrase is the node which "summarizes" its meaning (cf. Zwicky 1985). For instance, *un verre à vin* 'a wine glass' is a glass and its top node is *verre* 'glass', while *un verre de vin* 'a glass of wine' is wine and its top node is *vin* 'wine'. Note that in both cases *verre* 'glass' is the semantic governor of *vin* 'wine'.

Fig. 3. *The semantic representation (without grammemes) of (3)*

A relative clause modifies its antecedent and therefore communicatively depends on it, but there is generally no semantic dependency between the antecedent (here DAME 'woman') and the communicative top node of the relative clause (here PENSER 'think'). Consequently the communicative dependency must be represented by an own link (the light arrow without label). Moreover, the antecedent fills a semantic position in the relative clause (here as argument of MARI 'husband') and is consequently communicatively dominated by the top node of the relative clause. Our semantic description is based on [12] (see Fig. 3).

3 Syntactic Module

The formalism of our syntactic module is based on Nasr's formalism ([19], [20]), itself inspired by the TAG formalism ([7]). It is a *lexicalized grammar*, that is, a formal grammar that has the form of a lexicon: each lexical unit is associated with a set of elementary structures which describe its possible syntactic environments. The correspondence between a morphological and a syntactic representation is ensured by a combination of elementary structures associated with the lexical unit labeling both representations.

3.1 Elementary Structures

An elementary structure is a portion of a syntactic representation enriched with some features on the labeling. Each element, node or dependency, receives a feature `type`. Elements of `type:1` elements are said black and elements of `type:0`, white. Intuitively, black elements are the elements really introduced by the elementary structure, while white elements are requirements.

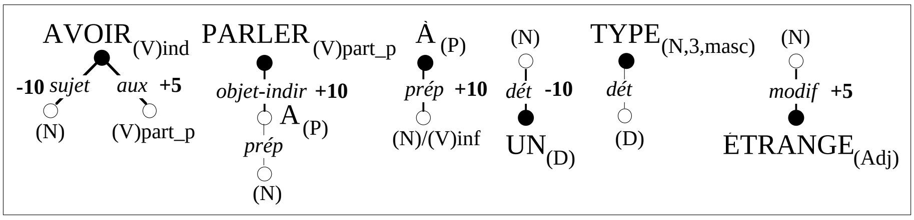


Fig. 4. *Elementary structures (AVOIR 'have', PARLER 'speak', À 'to', UN 'a', TYPE 'guy', ÉTRANGE 'strange')*

Moreover, each node a feature for the part of speech and some morphological features (gender, person…). Each dependency receives a feature `weight` for expressing the linear position of the dependent vis-à-vis its governor (see below). Some features can be not instantiated. (Black and white dependencies are represented by bold and light lines.)

3.2 Morphology-to-Syntactic Correspondence

Elementary structures are combined by merging of one or more elements; two elements can be merged if their labels can be unified. Note that the merging of a black and a white element gives a black element, while the merging of two white elements gives a white element; two black elements cannot be merged ($0 \cup 0 = 0$, $0 \cup 1 = 1$ and $1 \cup 1 =$ `failure`).

A morphological representation M and a syntactic representation S *correspond* to each other if the nodes of M and S are labeled by the same lexical units and if it is possible to associate with each lexical unit an elementary structure such that the combination of the elementary structures gives a structure X subsuming both M and S. A morphological string M is subsumed by X if the nodes of X can be ordered respecting both the projectivity[5] and the local order constraints given by the weights on the dependencies, assuming that the sign of the weight (- or +) indicates if the dependent is before or after the governor and the absolute value of the weight indicates the relative distance between the dependent and the governor (see [15], [4] for similar ideas).

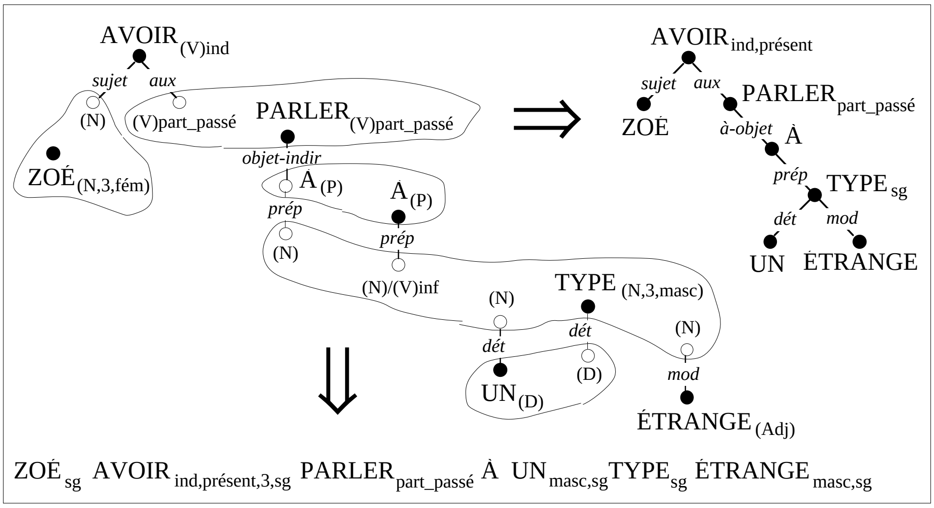


Fig. 5. *Combination of elementary structures and correspondence*

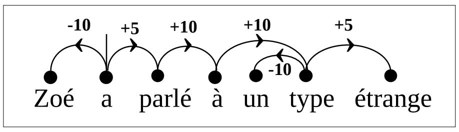


Fig. 6. *Projectivity and local order constraints*

Before seeing how our formalism can be enriched to take into account other phenomena, let us note that our formalism allows us to build non-lexicalized grammars equivalent to our lexicalized grammar. For instance, we can decide that an elementary

[5] A linearly ordered tree is said *projective* if no dependencies cross each other and no dependency covers the root.

structure contains only one black element; an elementary structure with a black node will be called *nodal* and an elementary structure with a black dependency, *sagittal* (lat. *sagitta* = arrow). A lexicalized elementary structure can be got back by merging of nodal and sagittal elementary structures (Fig. 7).

The partition of the rules into nodal and sagittal rules is equivalent to the partition of the grammar (= linguistic model) into lexicon and grammar (in the true sense). A nodal structure indicates the subcategorization pattern of a lexical unit, while a sagittal structure indicates the properties of a syntactic relation, such as linearization or agreement. Although our formalism is different from traditional MTT formalism ([18]), a sagittal structure is equivalent to an MTT surface syntactic correspondence rule.

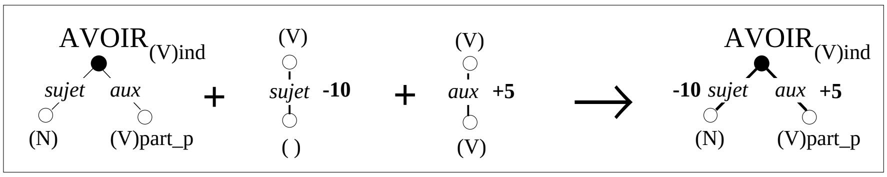


Fig. 7. *Combination of nodal and sagittal structures giving a lexicalized structure*

3.3 Quasi-Dependency

The biggest problem of all the lexicalized grammars we know (essentially TAG and the Categorial Grammars) is the combinatorial explosion of the number of structures associated with a lexical unit. This problem is generally due to a bad repartition of the linguistic information between the elementary structures. This repartition depends on the answer of questions such as the following: which of two words decides if they can combine? which decides how they are positioned towards each other? etc.

The first example we will study is the attributive and predicative use of an adjective. Compare *ce livre étrange* ‘this strange book’, *ce livre est étrange* ‘this book is strange’ et *Pierre trouve ce livre étrange* ‘Pierre finds this book strange’. In all these constructions, LIVRE ‘book’ is the semantic argument of ÉTRANGE ‘strange’. Moreover, in French, in all these cases the adjective must agree with the noun, which could be a problem for predicative adjectives, if they are not linked to it. Therefore, we will introduce in the elementary structure of a copulative verb a sort of dependency we will call a *quasi-dependency*. Quasi-dependencies are distinguished from real dependencies by the fact they have no role in the linearization (and consequently they do not bear a weight). Nevertheless, they bear a syntactic relation and a type. Formally, dependencies and quasi-dependencies will be differentiated by a + or - value of an ad hoc feature `quasi`. Quasi-dependencies are represented by broken arrows.

A *modif(icative)* quasi-dependency from the adjective to the noun modified is introduced in the elementary structure of copulative verbs. The same elementary structure can now be used for the attributive and predicative uses of an adjective, provided a dependency can merge with a quasi-dependency to give a dependency (Fig. 8).

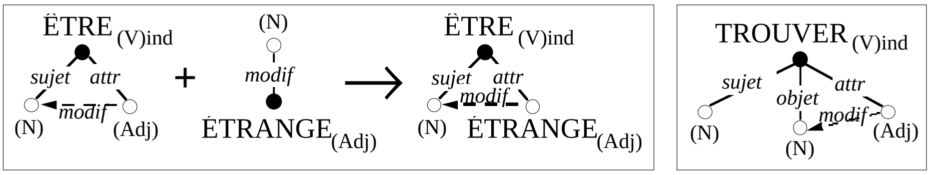


Fig. 8. *Combination of the copula (ÊTRE 'be') and the adjective (ÉTRANGE 'strange'). Elementary structure for the predicative verb(TROUVER 'find')*

The same ideas can be used for control and raising verbs (section 4). Quasi-dependencies will be also used for argument in non-canonical position (section 5).

4 Semantic Module

The semantic rules are of two types: lexical rules for the expression of deep lexical units and grammatical rules for the expression of deep grammemes.

A *lexical semantic rule* indicates the correspondence between a deep lexical unit and a configuration of surface lexical units. The correspondence between semantic actants and syntactic nodes is indicated by the variables *x*, *y*, etc. The deep grammemes that must be translated by grammatical rules are simply copied out in the syntactic configuration (right part of the rule) and preceded by an arrow (→) indicating that they must be translated by grammatical rules; for a French verb, it is the m(ood), the t(ense) and the v(oice). A special feature, symbolized by *, indicates that a node cannot be modified.

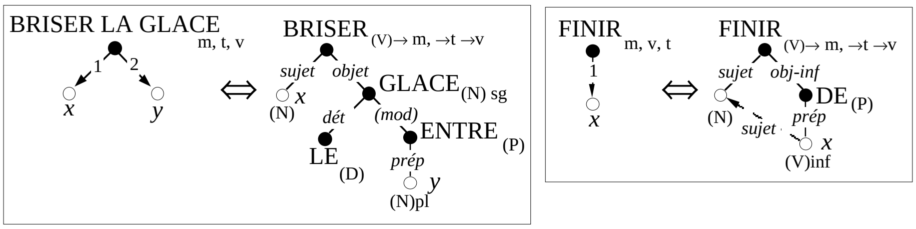


Fig. 9. *Lexical semantic rules for the idiom BRISER LA GLACE 'break the ice' and the raising verb FINIR 'finish'*

A *grammatical semantic rule* translates a deep grammeme into a surface grammeme or a more complex configuration. For instance, the inf(initive) mood is expressed by the surface grammeme inf and the transformation of the subject (Fr. *sujet*) dependency into a quasi-dependency. The passive voice (Fr. passif) is expressed by ÊTRE + -é, while the object (Fr. *objet*) node becomes subject and the subject node becomes an optional prepositional complement introduced by the prepositions PAR 'by' (in the default case). Note that the grammatical semantic rules are not correspondences between a semantic and a syntactic configuration, but between two syntactic configurations. The order in which the grammatical rules are applied does not matter; the notation (→)m corresponds to both →m and m depending to the fact that a rule translating m has been yet applied.

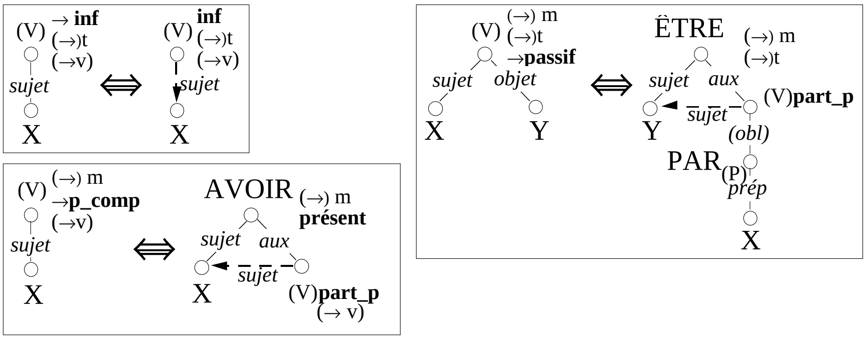


Fig. 10. *Grammatical semantic rule for the infinitive mood, the passive voice and the passé composé tense.*

5 Extractions

This section is devoted to the description of extractions (topicalization, relativization, interrogation, clefting…). Extractions are problematic for at least two reasons. First, the ordered dependency tree of a sentence with an extraction is generally not projective (Fig. 11). Second, since Ross' work ([22]), it is well known that the extractions are subject to a lot of constraints. These two problems can be solved simultaneously.

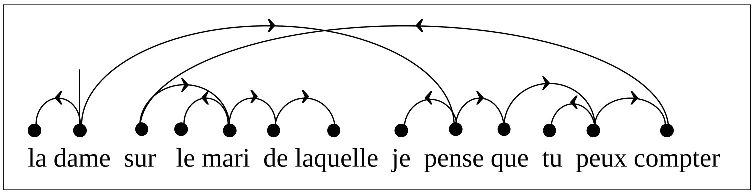


Fig. 11. *The (non projective) syntactic dependency tree of (3)*

5.1 Description of Extractions

Our description of extraction is based on a new linguistic concept, the *nucleus*, which has been introduced in [8] and used in [12]. Some groups of words, which we will call nuclei, can behave like a single word from the viewpoint of various phenomena including extraction. We consider two types of nuclei:

- A *verbal nucleus* is a string of verbs or equivalent forms (copula+adjective, light verb+predicative noun…): *un livre que Pierre* ***lit****,* ***a commencé à lire, a envie de lire, a l'air de trouver facile à lire****,* ***pense que*** *Zoé* ***devrait lire****…* 'a book that Pierre is reading, began to read, feels like reading, looks to find easy to read, thinks that Zoé should read…'
- A *nominal nucleus* is a string of nouns including prepositions and determiners: *une dame* ***sur laquelle*** *tu comptes,* ***sur le mari de laquelle*** *tu comptes…* 'a woman on whom you count, on whose husband you count…'

The exact nature of nuclei depends on the language: for instance, in English, contrary to French, a preposition can end a verbal nucleus (preposition stranding: *the girl you are talking to*, **la fille que tu parles à*). The nature of nuclei also depends on the linguistic phenomena: for instance, the strings *V que V* 'V that V', possible for extractions or gapping coordination, are more difficult for the negation: *la personne à qui Jean* ***veut que*** *tu* ***parles*** 'the person to who Jean wants that you speak'***; Jean veut qu'****on* ***appelle*** *la police et Zoé les pompiers* 'Jean wants that one calls the police and Zoé, the fire department; *??Jean ne* ***veut que*** *tu* ***parles*** *à personne* 'Jean does not want that you speak to anyone').

It is possible to integrate the nucleus in the syntactic representation (Fig. 12). Nuclei are represented by bubbles. The resulting structure is called a *bubble tree* ([8]). Bubbles trees are trees whose nodes can be filled by bubbles, which can contain others nodes which can, in their turn, be filled by bubbles or have their own dependents.

There are some advantages to consider nuclei and represent them by bubbles:

- Some phenomena can be accounted for more easily with the use of nuclei. For instance, relativization can be described as follows: 1) the wh-word must belong to a nominal nucleus governed by a verbal nucleus, itself governed by the antecedent of the relative; 2) the nominal nucleus must be positioned at the beginning of the relative clause. [6]
- While extraction clauses generally have a non-projective dependency tree, their bubble tree is projective[7]. The projectivity of the bubble tree allows us to ensure the linearization with only *local* linearization rules, that is, with weights on the dependencies.

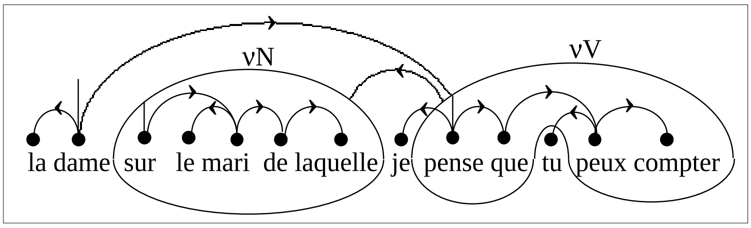


Fig. 12. *A (projective) bubble tree for (3)*

5.2 Elementary Structures for Relative wh-Words

In order to satisfy the constraints on extraction, a relative wh-word will introduce in its elementary structure a nominal nucleus and a verbal nucleus, the wh-word belonging to the nominal nucleus, the nominal nucleus depending on the verbal nucleus and the verbal nucleus depending on a nominal node. In other words, the whole information concerning the structure of a relative clause is encoded in the elementary

[6] Cases where the wh-word is not the top node of the nominal nucleus are called cases of *pied-piping*.

[7] A bubble tree is said *projective* if no dependencies and no bubbles cross each other and if no dependency and no bubble covers the top node.

structure of the wh-word, that is the constraints on extraction and the fact that a relative clause modifies a noun. We will now see how to solve the problem of the word order (and the non-projectivity).

In a lexicalized grammar such as TAG, actants are positioned in the elementary structure of their governor, whether they are in a canonical or non-canonical position.[8] This means a combinatorial explosion of the number of elementary structures. To avoid, this problem, in our grammar, only actants in canonical position are positioned in the structure of their governor. An actant in a non-canonical position will ensure its own positioning in its elementary structures. This can be done by introducing in the elementary structure associated with a lexical unit L in a non-canonical position a quasi dependency that will unify with the dependency between L and its governor and thus void the weight on it, L being positioned by another dependency bearing an appropriate weight. Such a situation occurs in the case of relativization, where the nominal nucleus (or, more exactly, its top node) is in a non-canonical position. Therefore the dependency between the verbal nucleus and the nominal nucleus in the elementary structure of a relative wh-word will in fact be a quasi-dependency. This will ensure that the top node of the nominal nucleus will not be positioned with respect to its syntactic governor. But how is the nominal nucleus positioned? A solution would be to add a dependency between the verbal nucleus and the nominal nucleus bearing a negative weight. We will adopt another solution, linguistically more motivated.

As Tesnière said, the relative wh-word, besides its pronominal role in the relative clause, has a subordinating role, allowing a clause to modify a noun. The relative wh-word transfers (Fr. *translater*) the verb into an adjective, that is, it masks the verb, allowing it to modify a noun and to occupy a syntactic position traditionally occupied by an adjective.[9] For this reason, Tesnière treats a wh-word as a split node, one part dominating the main verb of the clause, the other part filling a position in the clause.

[8] The term *position* refers to the linear position (in the morphological string). A position in the syntactic tree is called a syntactic position.

[9] The grammemes of past participle and present participle are other "translators" of verbs into adjectives: *le livre achet**é** par Pierre* 'the book bought by Pierre'; *la personne achet**ant** un livre* 'the person buying a book'. On the contrary, the infinitive and the subordinating conjunction *que* 'that' transfers the verb into a noun, allowing it to occupy an actantial position: *Pierre souhaite le départ de Marie/part**ir**/**que** Marie parte* 'Pierre wishes the departure of Marie/to leave/that Marie leaves'.

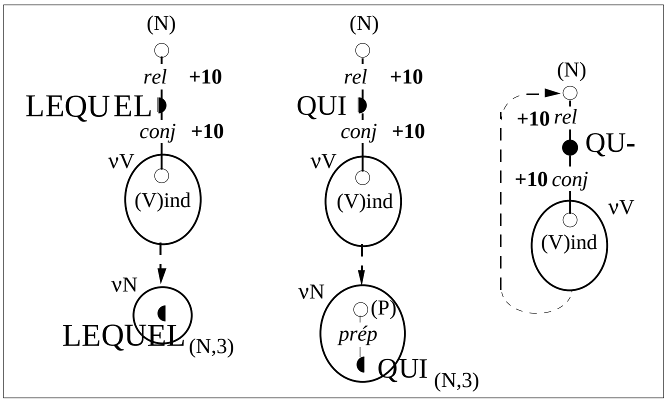


Fig. 13. *Elementary structures for LEQUEL 'which', QUI 'who' and QU- 'that'*

Our elementary structures reflect Tesnière's analysis, the wh-word being split into two semi-nodes. The linearization processes is as follows: the QU- part of the wh-word which dominates the verbal nucleus must be positioned before the verbal nucleus (due to the weight on the dependency linked them); the two part of the wh-word, which forms a single word, must be adjacent; the positioning of the nominal nucleus is conditioned by the positioning of the pronominal part of the wh-word, which belongs to it. Note that the elementary structure of a wh-word combines with three elementary structures: the antecedent noun, the syntactic governor of the wh-word in the relative clause (the governor of the "gap") and the main verb of the relative clause (moreover the wh-words constrains this verb to bear the indicative mood).

Some further information about French relative wh-words. First, the wh-word LEQUEL can be used in all syntactic positions, although the subject and direct object positions are only possible in non-restrictive relative clause (the combination À LEQUEL and DE LEQUEL give the contracted form *auquel* and *duquel*). Second, French has a parallel series of wh-word: QUI_1 for the subject position, QUE for the direct object position and QUI_2 after a preposition. Clearly, QUI_1 and QUI_2 are two different words because the second denotes only a human.

(4) a. *la personne/le chien qui dort*
'the person/the dog that is sleeping'
b. *la personne/le chien que tu regardes*
'the person/the dog that you are watching'
c. *la personne/*le chien à qui j'ai parlé*
'the person/the dog to who I spoke'

Some works since [13] seem to indicate that QUI_1 and QUE are two forms of a same wh-word, which we will call QU-, and that this wh-word is in fact the same word as the subordinating conjunction QUE (exactly like THAT in English). The best argument in favor of this hypothesis is the *qui/que* alternation in case of long distance dependencies:

(5) a. *la personne que je pense qui dort*
'the person that I think (that) is sleeping'

b. *la personne que je pense que tu regardes*
'the personne that I think that you are watching'

Indeed, contrary to what could be expected, in (5a), a case of long distance extraction of the subject, the relative wh-word (which as usual introduces the relative clause) is *que* since the subordinating conjunction is *qui* (which is very unusual, see (5b)). This alternation can be solved by saying that it is still the same word QU- which takes the *qui* form when the verb it subordinates has its subject extracted and takes the *que* form in all other cases (see a formal solution in TAG [10]).

5.3 Operations of Combination with Bubbles

The elementary structures with bubbles combine by unification of nodes and dependencies, like the other elementary structures. There is nevertheless a complication: what happens to the dependents of a node that is positioned in a bubble? Three cases are possible:1) the dependent goes outside the bubble (the default case), 2) the dependent goes inside the bubble as its governor or 3) the dependent is promoted to the bubble. Three different rules of combination, called R1, R2 and R3, will be introduced. Before presenting these rules, we will presents the mechanism which controls their possible application.

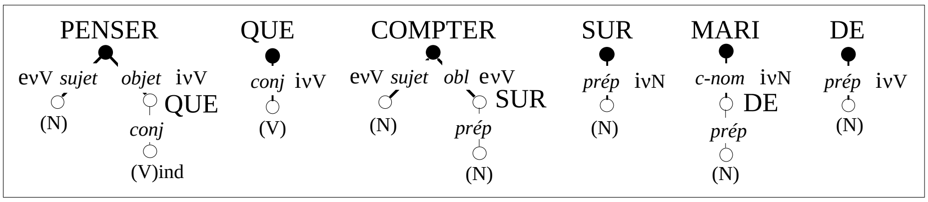


Fig. 14. *Elementary structure with labeling for intra- and extranuclear dependencies*

The fact that a dependency can be positioned inside a bubble or be promoted to a bubble is marked by the value of a feature `nucleus`. We have considered two types of bubbles, nominal and verbal nuclei, henceforth designated νN and νV. A dependency which can be inside a nucleus of type ν receives the `value nucleus:iν` (`i` for intranuclear). A dependency which can be promoted to a nucleus of type ν receives the `value nucleus:eν` (`e` for extranuclear) (Fig. 14).Now, when a node A' merges with a node A of a bubble ν, the nodes linked to A' are positioned by default outside ν (Rule R1, Fig. 15). However, the membership of ν can be propagated to whatever node B' linked to A' with an intranuclear dependency iν (Rule R2, Fig. 16). Finally, every node B' linked to A' with an extranuclear dependency eν can be promoted to the nucleus ν (Rule R3, Fig. 17).

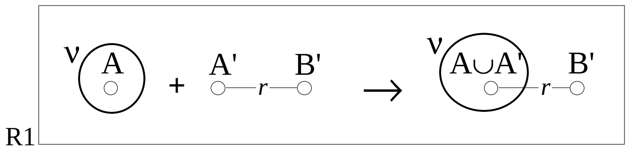


Fig. 15. *R1: Positioning by default of a dependent of a node of a nucleus ν*

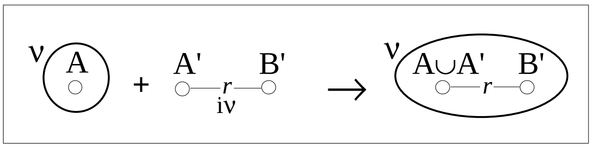
R2


Fig. 16. *R2: Propagation of the membership of a nucleus ν*

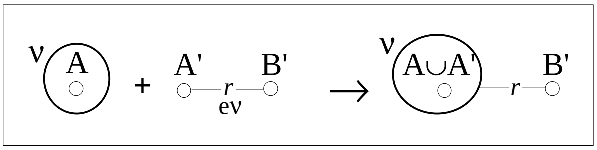
R3


Fig. 17. *R3: Promotion of a dependent on a nucleus ν*

We can now explain how the correspondence between the morphological and syntactic representations of (3) is obtained (Fig. 18). The only point that we think needs a comment is the *obl* dependency between COMPTER ‘count’ and SUR ‘on’. This dependency is labeled eνV, therefore when COMPTER is positioned in the νV bubble (= the verbal nucleus), it can be promoted to the νV bubble and then unify with the quasi dependency introduced by the elementary structure of LEQUEL.

We achieve a more or less exhaustive description of relativization in French. Other case of extraction cannot be developed here but we hope that we have convinced the reader that they can be described exactly in the same way.

5.4 Comparison with Other Formalisms

In our analysis, the nominal nucleus is promoted from its syntactic governor to the verbal nucleus. For instance, in the analysis of (3), the nominal nucleus *sur le mari de laquelle* does not depend on *compter* but on the verbal nucleus *pense que (tu) peux compter*. In terms of immediate constituents, we see that the projection of the verbal nucleus dominates directly the projection of the nominal nucleus (= the extracted phrase) and the projection of the main verb (the clause with a gap). Our analysis can

In A. Gelbukh (ed.), *Computational Linguistics*, 2001, Springer, 18-31.

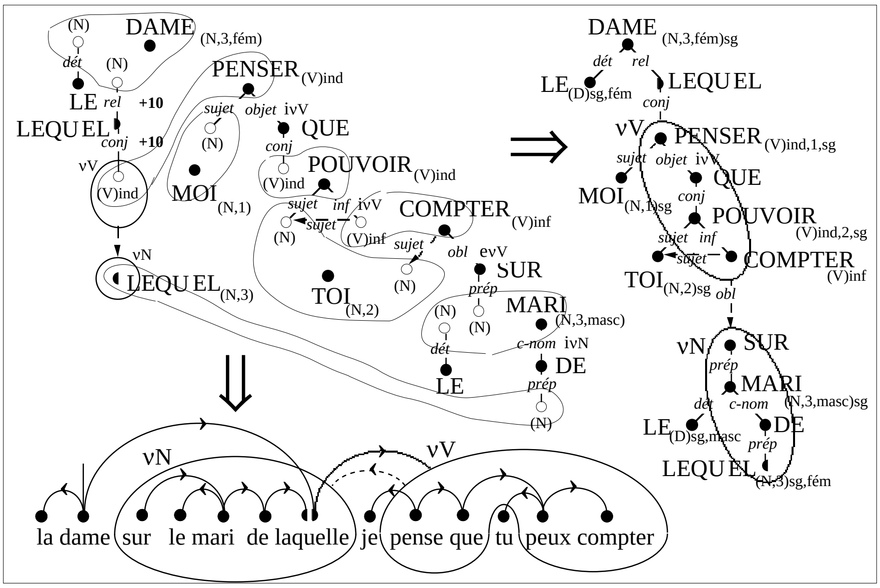


Fig. 18. *Combination of elementary structures for (3)*

therefore be compared of the X' Syntax's analysis ([3], [6]), where the relative clause corresponds to a node S' dominating a node COMP for the extracted phrase and a node S for the rest of the clause. Moreover, our operations can be compared to the movement of the X' Syntax or the slash technique of HPSG: the rule R3 is equivalent to the introduction of a non local (or slash) feature, while the rule R2 corresponds to the propagation of this feature. Several analyses in dependency grammar frameworks use devices equivalent to the slash feature, allowing lifting the extracted phrase on the main verb ([5], [14], [11]). Our analysis differs on at least one point: we do not propagate the content of the nominal nucleus (or extracted phrase) but we simply attach it to verbal nucleus and we propagate the membership of the verbal nucleus.

6 Conclusions

We have proposed a mathematical formalism for the implementation of grammars based on MTT. First, this formalism allows us to write fully lexicalized grammars, which are more efficient for the natural language processing, without bringing about the combinatorial explosion of the elementary structures. Second, our formalism can be extended in order to treat extractions, which are the corner stone of most of the formalisms developed since 25 years. This analysis is based on a linguistic concept, the nucleus, that we think plays a fundamental role in non-local phenomena such as extraction, coordination or negation. Moreover, this linguistic concept is not simply superimposed to the traditional syntactic representation, but it is encoded as a primitive concept, the bubble, associated to a set of geometric properties, such as the projectivity of bubble trees.

References

1. Abeillé Anne, *Une grammaire lexicalisée d'Arbres Adjoints pour le français*, Thèse de Doctorat, Univ. Paris 7 (1991).
2. Candito Marie-Hélène & Kahane Sylvain, "Can the TAG Derivation Tree Represent a Semantic Graph ? An answer in the Light of Meaning-Text Theory", TAG+4 Workshop, Philadelphie (1998) 21-24.
3. Chomsky Noam, "On Wh-movement", in P. Culicover et al. (eds), *Formal Syntax*, Dordrecht : Reidel (1977).
4. Courtin Jacques & Genthial Damien, "Parsing with Dependency Relations and Robust Parsing", in Kahane S. & Polguère A. (eds*), Workshop on dependency-based grammars, COLING-ACL'98*, Montréal (1998) 95-101.
5. Hudson Richard A., "Discontinuity", in S. Kahane (ed*.), Grammaires de dépendance, T.A.L., 41.1* (2000) 15-56.
6. Jackendoff Ray S., *X' Syntax : A Study of Phrase Structure*, Linguistic Inquiry Monograph, MIT Press (1977).
7. Joshi Aravind K., "Introduction to Tree Adjoining Grammar", in Manaster Ramer (ed.), *The Mathematics of Language*, Amsterdam:Benjamins (1987) 87-114.
8. Kahane Sylvain, "Bubble Trees and Syntactic Representations", in Becker T. & Krieger H.-U. (eds), *Proc. MOL'5*, Saarbrücken : DFKI (1997) 70-76.
9. Kahane Sylvain, "How to solve some failures of LTAG", *TAG+5 Workshop*, Paris (2000) 123-28.
10. Kahane Sylvain, Candito Marie-Hélène, de Kercadio Yannick, "An alternative description of extractions in TAG", *TAG+5 Workshop*, Paris (2000) 115-22.
11. Kahane Sylvain, Nasr A. & Rambow Owen, "Pseudo-Projectivity: a Polynomially Parsable Non-Projective Dependency Grammar", *COLING-ACL'98*, Montréal (1998) 646-52.
12. Kahane Sylvain & Mel' uk Igor, "La synthèse des phrases à extraction en français contemporain. Du réseau sémantique à l'arbre syntaxique", *T.A.L, 40 : 2* (1999) 25-85.
13. Kayne Richard, *French Syntax : the Transformational Cycle*, Cambridge : MIT Press (1975).
14. Lombardo Vincenzo & Lesmo Leonardo, "Formal Aspects and Parsing Issues of Dependency Theory", *COLING-ACL'98*, Montréal (1998) 787-93.
15. Mel' uk Igor, "Ordre des mots en synthèse automatique des textes russes", *T.A. Information, 2* (1967) 65-84.
16. Mel' uk Igor, *Dependency Syntax: Theory and Practice*, NY : SUNY Press (1988).
17. Mel' uk Igor, "Paraphrase et lexique: la Théorie Sens-Texte et le Dictionnaire Explicatif et Combinatoire", in Mel' uk et al., *Dictionnaire Explicatif et Combinatoire*, *3* (1992) 10-57.
18. Mel' uk Igor & Pertsov Nicolas, *Surface syntax of English – A Formal Model within the Meaning-Text Framework*, Amsterdam : Benjamins (1987).
19. Nasr Alexis), "A Formalism and a Parser for Lexicalised Dependency Grammars", *4th Int. Workshop on Parsing Technologies*, SUNY Press (1995).
20. Nasr Alexis, *Un modèle de reformulation automatique fondé sur la Théorie Sens-Texte – Application aux langues contrôlées*, Thèse de Doctorat, Univ. Paris 7 (1996).
21. Polguère Alain, *Structuration et mise en jeu procédurale d'un modèle linguistique déclaratif dans un cadre de génération de texte*, Thèse de Doctorat, Univ. Montréal (1990).
22. Ross John, *Constraints on Variables in Syntax*, PhD Thesis, MIT (1967); *Infinite syntax !*, Dordrecht : Reidel (1985).
23. Tesnière Lucien, *Eléments de syntaxe structurale*, Paris: Kliencksieck (1959).
24. XTAG Research Group, *A Lexicalized TAG for English*, Technical Report IRCS 95-03, Univ. of Pennsylvania (1995) updated version on the web.