



Towards a DRS Parsing Framework for French

Luyen Le Ngoc, Yannis Haralambous, Philippe Lenca

► To cite this version:

Luyen Le Ngoc, Yannis Haralambous, Philippe Lenca. Towards a DRS Parsing Framework for French. SNAMS 2019 : Sixth International Conference on Social Networks Analysis, Management and Security, Oct 2019, Granada, Spain. 10.1109/SNAMS.2019.8931868 . hal-02280800

HAL Id: hal-02280800

<https://hal.science/hal-02280800>

Submitted on 6 Sep 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Towards a DRS Parsing Framework for French

Luyện Ngọc Lê
IMT Atlantique
UMR CNRS 6285 Lab-STICC
CS 83818,
29238 Brest Cedex 3, France
luyen.le@imt-atlantique.fr
ORCID: 0000-0002-2043-0679

Yannis Haralambous
IMT Atlantique
UMR CNRS 6285 Lab-STICC
CS 83818,
29238 Brest Cedex 3, France,
yannis.haralambous@imt-atlantique.fr
ORCID: 0000-0003-1443-6115

Philippe Lenca
IMT Atlantique
UMR CNRS 6285 Lab-STICC
CS 83818,
29238 Brest Cedex 3, France,
philippe.lenca@imt-atlantique.fr
ORCID: 0000-0001-5014-2347

Abstract—Combinatory Categorical Grammars provide a transparent interface between surface syntax and underlying semantic representation. Discourse Representation Theory allows the handling of meaning across sentence boundaries. Based on the foundations of these two theories along with the work of Johan Bos on the Boxer framework for English language, we propose an approach to the task of semantic parsing with Discourse Representation Structure for the French language. By giving an example of discourse analysis on French sentences and experimenting on 4,525 sentences taken from the French Treebank corpus, we demonstrate and evaluate the outcomes of our framework.

Index Terms—Semantic Parsing, Discourse Representation Structure, Combinatory Categorical Grammars, Boxer, French language.

I. INTRODUCTION

Researchers in the domain of computational linguistics have studied for a long time various forms of logic aiming to capture semantic information from natural language data. Among these, Discourse Representation Theory (DRT) is one of the first frameworks for exploring meaning across sentences, with a formal semantics approach [1]. DRT has been used for various applications such as the implementation of a semantic parsing system or natural language understanding systems. This paper addresses the build of a semantic parsing application based on Combinatory Categorical Grammars (CCG) and Discourse Representation Structure (DRS) which is an important element of DRT.

CCGs essentially illustrate an explicit relation between syntax and semantic representation [2]. They allow access to a deep semantic structure of the phrase and facilitate recovering of non-local dependencies involved in the construction, such as coordination, extraction, control, and raising. In addition, CCGs are compatible with first order logic (FOL) and lambda-expressions. Their use allows for analyzing the syntax and semantic relationship between words or phrases in the scope of the sentence.

DRT can be used to deal with a variety of linguistic phenomena such as anaphoric pronouns or temporal expressions within or across different sentences. DRS expressions have two advantages: (1) they provide meaning representation for several sentences in discourse instead of a single sentence only, and (2) they can be translated into FOL.

The main contribution of our work is the construction of a discourse parsing framework for the French language based on the Boxer framework by Johan Bos [3]. For each discourse input, we perform semantic parsing and generate a formal meaning representation in the form of DRS or FOL output.

In this article, after an introduction to semantic parsing, we provide an overview of the latest research for building a semantic parser for French. In the next section, we present the theoretical background of CCG and DRT. Then we describe our approach, experimentation, and evaluation through an example. Finally, we finish the paper with a conclusion and ideas for future work.

II. RELATED WORKS

Semantic parsing is formally the task of mapping a natural language expression to one or many machine-interpretable meaning representations. There exist mainly two approaches for building a semantic parser using the DRT framework for semantic representation. The first one is the empirical approach based on the domain of symbolic logic [4]–[6]. The second one uses neural networks along with their variants such as sequence models [7] or encoder-decoder models [8].

An outstanding work based on the empirical approach is the *Boxer* system by Johan Bos and his colleagues [3], [6], [9]. Used as an open-domain semantic parser, Boxer requires in the input syntactically analyzed data under the form of CCG derivations, each sentence corresponding to a single CCG derivation. For English, one can obtain CCG derivations by using a CCG parser such as C&C Parser [10], EasyCCG [11], or OpenCCG¹. The output of Boxer is a meaning representation in the form of DRS in XML or boxed format (see Fig. 6).

Concerning French language, there exists a tool called *Grail* which allows parsing of French discourses in order to obtain meaning representations in DRS [12]–[15]. Unlike *Boxer*, the input of *Grail* requires syntactically analyzed data based on the Type-Logic Grammar (TLG) formalism [16], [17]. TLG has been mostly applied in theoretical issues and relations to logic and theorem proving, while CCGs have been

¹<https://github.com/OpenCCG/openccg>

more concerned with keeping expressive power and automata-theoretic complexity to a minimum. Therefore, CCGs are more relevant to the issues of linguistic explanation and practical computational linguistic [18].

In the last decade, due to the emergence of neural network models and their proven efficiency in a variety of NLP tasks, people have investigated handling semantic parsing as a sequence transduction problem where discourse is mapped into a meaning representation format [19]–[22]. At the same time, the release of meaning representation corpora such as *Groningen Meaning Bank* [23] and *The Parallel Meaning Bank* [24] has opened new research directions in the development of semantic parsing applications. However, the results are still limited and strongly dependent on training corpora.

In order to build a French semantic parser based on CCGs and DRT, a resource lacking until today, we chose the empirical approach, adapted to French language. We describe it in detail in the next section.

III. THEORETICAL BACKGROUND

A. Combinatory Categorical Grammars

Combinatory Categorical Grammars have been introduced by Mark Steedman [2], [25] as a non-transformational grammatical theory relying on combinatory logic. The notion of CCG is an extension of AB Categorical Grammar, as introduced in [26] and [27].

Formally, a CCG is a set $G = \{\Sigma, \Delta, f, \varsigma, \mathcal{R}\}$, where:

- Σ defines a finite set of *terminals*.
- Δ defines a finite set of *axiom categories*, e.g., S, NP, N, PP etc. More complex categories are created by using the backward \backslash and forward $/$ operators, e.g., if $X, Y \in \Delta$, then X/Y and $X \backslash Y \in \Delta$.
- f is a *lexical category function* mapping terminals into categories.
- ς is the unique *starting symbol*, $\varsigma \in \Delta$.
- \mathcal{R} is a finite set of *combinatory rules* which we describe below.

By X, Y, Z we denote “meta-categories” which stand for any possible category. The set \mathcal{R} includes two basic rules inherited from AB categorial grammars:

$$\begin{array}{lcl} \text{Forward Application:} & & \\ X/Y:f & Y:a & \Rightarrow X:f(a), \\ \text{Backward Application:} & & \\ Y:a & X \backslash Y:f & \Rightarrow X:f(a). \end{array}$$

An important contribution of CCG theory is the extension of AB categorial grammars by a set of rules based on the *combinators* of combinatory logic [28]. These rules allow processing of long-range dependencies and of extraction/coordination constructions. They can be represented via λ -calculus, used as a notation for semantic representation, as follows:

$$\begin{array}{lcl} \text{Forward Composition:} & & \\ X/Y & Y/Z & \Rightarrow X/Z \\ \lambda y.f(y) & \lambda z.g(z) & \Rightarrow_B \lambda z.f(g(z)) \end{array}$$

Forward Crossing Composition:

$$\begin{array}{lcl} X/Y & Y \backslash Z & \Rightarrow_B X \backslash Z \\ \lambda y.f(y) & \lambda z.g(z) & \Rightarrow \lambda z.f(g(z)) \end{array}$$

Backward Composition:

$$\begin{array}{lcl} Y \backslash Z & X \backslash Y & \Rightarrow_B X \backslash Z \\ \lambda z.g(z) & \lambda y.f(y) & \Rightarrow \lambda z.f(g(z)) \end{array}$$

Backward Crossing Composition:

$$\begin{array}{lcl} Y/Z & X \backslash Y & \Rightarrow_B X/Z \\ \lambda z.g(z) & \lambda y.f(y) & \Rightarrow \lambda z.f(g(z)) \end{array}$$

Forward Substitution:

$$\begin{array}{lcl} (X/Y)/Z & Y/Z & \Rightarrow_S X/Z \\ \lambda zy.f(z, y) & \lambda z.g(z) & \Rightarrow \lambda z.f(z, g(z)) \end{array}$$

Forward Crossing Substitution:

$$\begin{array}{lcl} (X/Y) \backslash Z & Y \backslash Z & \Rightarrow_S X \backslash Z \\ \lambda zy.f(z, y) & \lambda z.g(z) & \Rightarrow \lambda z.f(z, g(z)) \end{array}$$

Backward Substitution:

$$\begin{array}{lcl} Y \backslash Z & (X \backslash Y) \backslash Z & \Rightarrow_S X \backslash Z \\ \lambda z.g(z) & \lambda zy.f(z, y) & \Rightarrow \lambda z.f(z, g(z)) \end{array}$$

Backward Crossing Substitution:

$$\begin{array}{lcl} Y/Z & (X \backslash Y)/Z & \Rightarrow_S X/Z \\ \lambda z.g(z) & \lambda zy.f(z, y) & \Rightarrow \lambda z.f(z, g(z)) \end{array}$$

There are also type-raising rules in CCG:

Forward type-raising:

$$X:x \Rightarrow_T T/(T \backslash X): \lambda f.f(x)$$

Backward type-raising:

$$X:x \Rightarrow_T T \backslash (T/X): \lambda f.f(x)$$

In the case of coordination constructions, the lexical category $(X/X) \backslash X$ is used to validate the combination of similar components in the rules of formulas above. It is formalized as follows:

Coordination:

$$X:g \quad X:f \Rightarrow_{\Phi^n} X:\lambda x.f(x) \wedge g(x)$$

Let us give an example: we will apply the definition of CCG to $G = \{\Sigma, \Delta, f, \varsigma, \mathcal{R}\}$, where:

- $\Sigma = \{\text{Henri, regarde, la, télévision}\}$.
- $\Delta = \{S, NP, S \backslash NP\}$
- $f: f(\text{Henri}) = \{NP, NP/NP\}$, $f(\text{regarde}) = \{S \backslash NP, (S \backslash NP)/NP\}$, $f(\text{la télévision}) = \{NP\}$
- $\varsigma = S$ (sentence)
- \mathcal{R} as defined in the formulas above.

There are two different CCG derivations for the sentence “Henri regarde la télévision” (Henri watches television) based on the lexicon that was defined above. A sentence can be parsed into many different CCG structures. In other words, we encounter a situation where a grammatical sentence may have many valid and equivalent semantic parses. This phenomenon is called *spurious ambiguity* and is the cause of a combinatory explosion in the search space when we parse. The difference between the two examples below is that in the second case we use type raising for “Henri” and for “la télévision”:

CCG derivation 1:

$$\begin{array}{c}
\text{Henri} \quad \text{regarde} \quad \text{la} \quad \text{télévision} \\
\overline{NP} \quad \overline{(S \backslash NP) / NP} \quad \overline{NP / NP} \quad \overline{NP} \\
\hline
: x \quad : \lambda x \lambda y. \text{regarde}' xy \quad : y \\
\hline
\overline{S \backslash NP : \lambda x. \text{regarde}' la_télévision' x} \\
\hline
\overline{S : \text{regarde}' la_télévision' henri'}
\end{array}$$

CCG derivation 2:

$$\begin{array}{c}
\text{Henri} \quad \text{regarde} \quad \text{la} \quad \text{télévision} \\
\overline{NP} \quad \overline{(S \backslash NP) / NP} \quad \overline{NP / NP} \quad \overline{NP} \\
\hline
: x \quad : \lambda x \lambda y. \text{regarde}' xy \quad : y \\
\hline
\overline{S / (S \backslash NP)}^{\tau} \quad \overline{S \backslash (S / NP)}^{<\tau} \\
: \lambda p. p \text{ henri}' \quad : \lambda p. p la_télévision' \\
\hline
\overline{S / NP : \lambda x. \text{regarde}' x \text{ henri}'}^B \\
\hline
\overline{S : \text{regarde}' la_télévision' henri'}
\end{array}$$

B. Discourse Representation Structure

One of the first theoretical frameworks for dynamic semantics has been introduced by Hans Kamp under the name *Discourse Representation Theory* (DRT) in order to deal with linguistic phenomena such as anaphoric pronouns, time, tense, presupposition, and propositional attitudes [1]. Essentially, DRT uses a semantic representation, called “Discourse Representation Structure” (DRS) $\langle drs \rangle$ which includes two principal components: a set of discourse referents $\langle ref \rangle$ which represent the objects under discussion and a set of conditions $\langle condition \rangle$ which are properties of discourse referents, and express relations between them. In general, DRS conditions are of two types: $\langle basic \rangle$ and $\langle complex \rangle$:

$$\langle drs \rangle ::= \frac{\langle ref \rangle^*}{\langle condition \rangle^*} \quad (1)$$

$$\langle condition \rangle ::= \langle basic \rangle | \langle complex \rangle \quad (2)$$

The basic conditions are properties of discourse referents or relations between them:

$$\begin{aligned}
\langle basic \rangle ::= & \langle sym_1 \rangle(\langle exp_e \rangle) | \langle sym_2 \rangle(\langle exp_e \rangle, \langle exp_e \rangle) | \\
& \langle exp_e \rangle = \langle exp_e \rangle | \langle exp_e \rangle = \langle num \rangle | \\
& time_x(\langle exp_e \rangle, \langle sym_0 \rangle) | \\
& named(\langle exp_e \rangle, \langle sym_0 \rangle, class)
\end{aligned} \quad (3)$$

where $\langle exp_e \rangle$ denotes expressions of type, $\langle sym_n \rangle$ describe n -place predicates, $\langle num \rangle$ denotes cardinal numbers, $time_x$ expresses temporal information and $class$ denotes named entity classes.

The complex conditions represent embedded DRSs: implication(\Rightarrow), negation (\neg), disjunction (\vee), modal operators

expressing necessity (\Box) and possibility(\Diamond). The types of complex conditions are unary and binary:

$$\begin{aligned}
\langle complex \rangle & ::= \langle unary \rangle | \langle binary \rangle \\
\langle unary \rangle & ::= \neg \langle exp_t \rangle | \Box \langle exp_t \rangle | \Diamond \langle exp_t \rangle | \langle ref \rangle : \langle exp_t \rangle \\
\langle binary \rangle & ::= \langle exp_t \rangle \rightarrow \langle exp_t \rangle | \langle exp_t \rangle \vee \langle exp_t \rangle | \\
& \quad \langle exp_t \rangle ? \langle exp_t \rangle
\end{aligned} \quad (4)$$

where the condition $\langle ref \rangle : \langle exp_t \rangle$ denotes verbs with propositional content.

Let us dive into DRS through an example: “La femme achète des poissons” (the woman buys fishes) (i). This sentence can be analyzed and be rewritten under DRS form as follows: $[x, y : femme(x), poisson(y), acheter(x, y)]$. More specifically, the DRS expression contains two discourse referents x and y , and the set of conditions is $\{femme(x), poisson(y), acheter(x, y)\}$. Suppose now that the sentence (i) is followed by the sentence “Elle les donne à son mari.” (She gives them to her husband) (ii). The discourse will be rewritten as a semantic representation $[u, v, w : donner(u, v, w), mari(w)]$. Finally, we obtain the merge of the sentences (i) and (ii): $[x, y, u, v, w : v = x, w = y, femme(x), poisson(y), acheter(x, y), donner(u, v, w), mari(u)]$.

In order to illustrate the different DRS expressions, we have three ways of representing the above sentences:

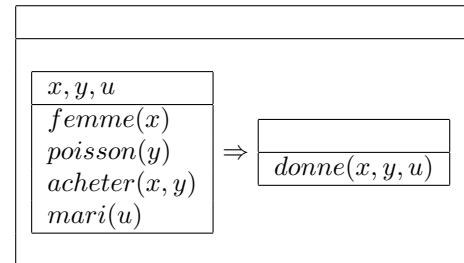
(1) The “official” DRS notation:

$$\begin{aligned}
& \langle \{ \}, \langle \{x, y, u\} \rangle, \{femme(x), poisson(y), \\
& \quad acheter(x, y), mari(u)\} \rangle \\
& \Rightarrow \langle \{ \}, \{donne(x, y, u)\} \rangle.
\end{aligned}$$

(2) The linear notation:

$$\begin{aligned}
& [: [x, y, u : femme(x), poisson(y), \\
& \quad acheter(x, y), mari(u)] \Rightarrow [: donne(x, y, u)]
\end{aligned}$$

(3) The boxed notation:



DRS expressions are compatible with first-order logic (FOL) through specific steps of translation [29], [30]. Firstly, we consider each discourse referent as a first-order quantifier. Then, DRS conditions are interpreted into a conjunctive formula of FOL. Finally, the embedded DRSs such as implication, negation, disjunction are translated to corresponding formulas [3].

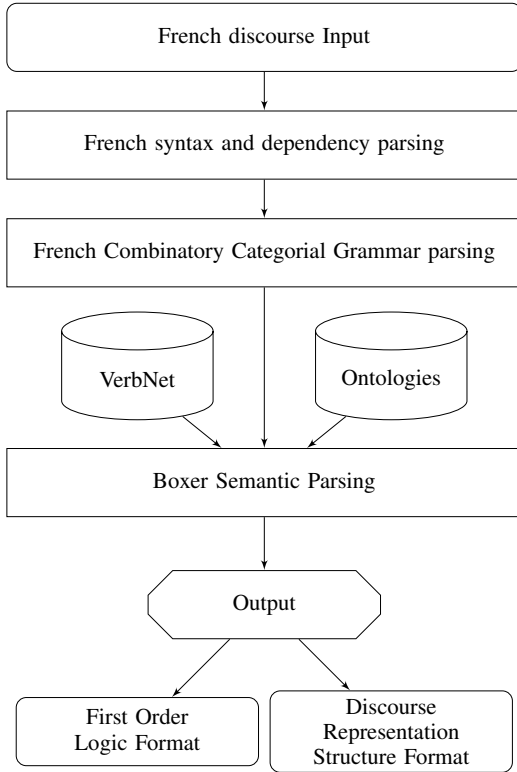


Fig. 1: Proposed Architecture for French Semantic Parsing

IV. PROPOSED ARCHITECTURE

In this section, we will describe the main components of the architecture we propose for semantic parsing (Figure 1).

A. Syntax and Dependency parsing

In general, a dependency structure of sentences is a result of the analysis and description of the dependency of the linguistic units, e.g., words, which are connected to each other by dependency paths. Most of the time, the root of the dependency tree corresponds to the main verb and all other words are either directly or indirectly connected to this verb by directed links. Each link has a label for describing the relation between the two words. These labels belong to a set of syntactic functions, e.g., subject, object, oblique, determiner, attribute, etc. Syntactic functions are grammatical relations playing an important role in recognizing components of the sentence.

For the input discourse, the syntactic information on words and their interrelations can be obtained via a dependency parser. There exist nowadays various dependency parsers for French such as MaltParser [31], Stanford Parser [32], SpaCy, and Grew Parser [33]. We have chosen to use MaltParser in order to obtain morphosyntactic information on words in sentences. We keep the following information for every word: lemma, part-of-speech tag, and dependency relation.

B. Combinatory Categorical Grammar Parsing

In order to obtain a CCG derivation tree for each input French sentence, we have used the empirical approach introduced in [34] and briefly explained below. Using the syntax and dependency information obtained in the previous step, we process words which have unique lexical categories, e.g., nouns have lexical category NP, adjectives have lexical category NP/NP or NP\NP depending on whether they are on the left of or on the right of the noun, etc. Once we have assigned these unique (but position-dependent, since, for example, adjectives in French can be located on both sides of the noun) lexical categories, we move over to verbs.

The main verb of the sentence, which is normally the root of the dependency tree, may have *argument dependencies*, labeled *suj*, *obj*, *a_obj*, *de_obj*, *p_obj*, i.e., correspondences with subject, direct and indirect object, and/or *adjunct dependencies* labeled *mod*, *ats*, etc., representing complementary information such as number, time, place, etc. The lexical category $S \backslash NP$ is assigned to a main verb having a subject to its left, and then we add a $/NP$ (or a $\backslash NP$, depending on its position with respect to the verb) for each direct object or indirect object (in the order of words in the sentence).

The next step is to binarize the dependency tree on the basis of information about dominant sentence structure: In French, most sentences are SVO, as in “*Mon fils (S) achète (V) un cadeau (O)*” (my son buys a gift), or SOV as in “*Il (S) le (O) donnera (V) à sa mère (indirect O)*” (he will give it to his mother). Using this general linguistic property, an algorithm has been proposed in [34] to extract and classify the components of the sentence into: subject, direct object, indirect object, verbs, complement phrases. This algorithm aims to transform a dependency tree into a binary tree. It is subdivided into two steps:

- 1) Chunks are extracted from the dependency tree based on syntactic information and dependency labels between words. For example, the subject chunk is obtained by finding a word that has a dependency labeled *suj*; the verb chunk corresponds to the root of the dependency structure; direct or indirect object chunks are obtained as words with links directed to the root verb and having labels *obj* or *p_obj*, etc.
- 2) A binary tree is built for each chunk, and then binary trees are combined in inverse order of the dominant sentence structure. For example if SVO is the dominant structure, we start by building the binary tree of the *object* chunk, then combine it with the binary tree of the *verb* chunk, and finally we obtain the binary tree of the *subject* chunk.

For each input sentence we obtain a single CCG derivation tree, corresponding to its dependency tree input. The output CCG derivation tree is modified to be compatible with the *Boxer*’s input format. At the same time, the sentence is analyzed in order to extract named entity components (e.g., *Location*, *Person*, *Date*, *Time*, *Organization*, etc.) and chunk phrases by using the French models of SpaCy application.

% Primary: **'NP V NP'** ('allow-64')
 % Syntax: [np:'**Agent**',v,np:'**Theme**']
 % CCG: (s:dcl\np)/np
 % Roles: [**'Theme'**,**'Agent'**]
 % Example: **'Luc approuve l'attitude de Léa'**

VerbNet:
 (approuver, (s:dcl\np)/np, [**'Theme'**,**'Agent'**]).
 (autoriser, (s:dcl\np)/np, [**'Theme'**,**'Agent'**]).
 (supporter, (s:dcl\np)/np, [**'Theme'**,**'Agent'**]).
 (tolérer, (s:dcl\np)/np, [**'Theme'**,**'Agent'**]).

Fig. 2: An excerpt of the Verbnet lexical resource for French

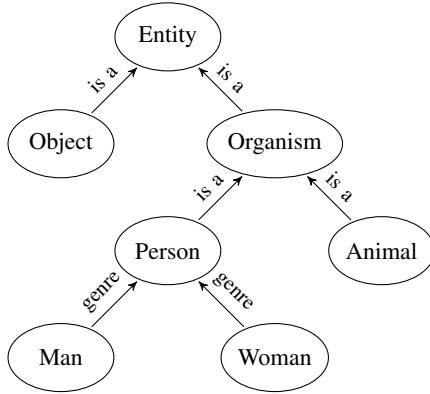


Fig. 3: An ontology resource example

C. Boxer Semantic Parsing

Implemented in the Prolog language with publicly available source code, the *Boxer* application is designed to provide semantic analysis of discourses for English with CCG derivation trees as input and meaning representation under the form of DRS as output. In order to do the same in French, we had to adapt the source code to the specific characteristics of the French language.

Verbs are the central component of any sentence. Once a verb is given, we are able to know the components that can be attached to it. For example, the verb “to buy” must be followed by a direct object, and the verb “to sleep” cannot since it is intransitive. Relationships between a verb and its noun phrase arguments are illustrated by thematic roles (e.g., Agent, Experience, Theme, Goal, Source, etc.). In *Boxer*, verbs and their thematic roles are extracted from the VerbNet lexical resources corpus [35]. For French, we have used the French VerbNet corpus [36] (see an example in Fig. 2), while ontologies have provided hierarchical or equivalent relationships between entities, concepts, etc. (Fig. 3).

Issues concerning anaphora and presupposition triggers introduced by noun phrases, personal pronouns, possessive pronouns, reflexive pronouns, demonstrative pronouns, etc., are processed on a case-by-case basis, based on the resolution



Fig. 4: CCG derivation tree for the sentence “Tous les soirs, mon voisin met sa voiture au garage”.

$\exists z3 z7 z8 z9 z10 z11 z12 z13 z14 z5 z6. (np_fils(z6) \wedge de(z6, z5) \wedge np_male(z5) \wedge np_rosier(z14) \wedge de(z14, z13) \wedge np_male(z13) \wedge np_male(z12) \wedge au(z10, z11) \wedge np_garage(z11) \wedge np_voiture(z10) \wedge de(z10, z9) \wedge np_male(z9) \wedge np_voisin(z8) \wedge de(z8, z7) \wedge np_male(z7) \wedge \forall z4. (np_soir(z4) \rightarrow \exists z1. (a_topic(z4) \wedge \exists z2. (Recipient(z2, z10) \wedge Theme(z2, z8) \wedge a_mettre(z2)) \wedge alors(z4, z1))) \wedge Theme(z3, z14) \wedge Actor(z3, z12) \wedge a_arroser(z3) \wedge avec(z14, z6))$

Fig. 5: FOL output of the utterance

algorithm proposed in [37]. Finally, the meaning representation of the discourse analysis is exported in two different formats: FOL and DRS.

V. EXPERIMENTS

We illustrate the capabilities of French discourse analysis via our architecture by the following example: “Tous les soirs, mon voisin met sa voiture au garage. Il arrose ses rosiers avec

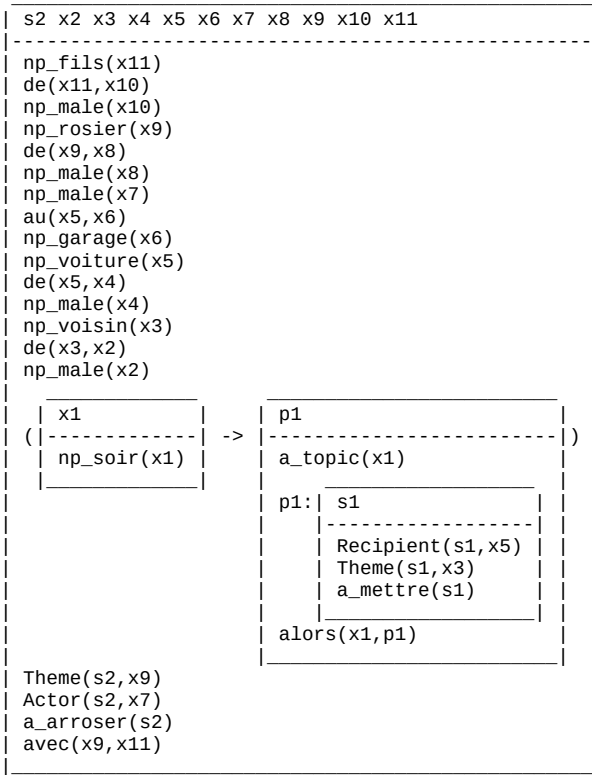


Fig. 6: DRS output of the utterance

son fils.” (Every evening, my neighbor puts his car in the garage. He waters his rose bushes with his son). We have two sentences in this text, containing possessive pronouns, personal pronouns, and noun phrases.

We first apply a parser to obtain dependency relations. We then obtain a CCG derivation tree as output of the CCG parsing stage. The results are represented (cf. Fig. 4) in a format which is compatible with the input format of Boxer. Each word is handled as a term (t) together with the following information: CCG lexical category, original word, lemma, POS tag label, chunks and named entity information.

The reader can see the output of the example in two formats: FOL (Figure 5) and DRS (Figure 6). Boxer for French can analyze correctly linguistic phenomena such as possessive pronouns (*ses, mon, sa, son*), propositional quantifiers (*tout*) and noun phrases (*sa voiture au garage*). However, there is still room for improvement, for example, we do not obtain the chronological order of actions in the example.

On the other hand, we experimented our system with 4,525 sentences from the French TreeBank corpus [38] in order to have an overview on a wide-coverage corpus. The length of the sentences in our experimentation is limited to 20 words because the FTB corpus was extracted from French newspapers, the sentences are thus regularly long and complex compared to the simple and short sentences of a discourse. Finally, we have obtained 61,94% of sentences which can be analyzed successfully by our system. By analyzing errors that occurred in our outcomes, we figure out two main causes.

The first one derives from errors in dependency analysis or CCG analysis step. The second one originates from the lack of semantic representation definition on the CCG lexical in Boxer for French.

Semantic parsing is a difficult task in the natural language processing field. We obtain a parse of French discourse step by step, and in order to obtain the semantic representation accurately, we have to ensure accuracy of previous analysis stages. If there is an error in them, this will fatally lead to errors in the results. For example, incorrect POS tags are one of the leading causes of erroneous results. Also phrases can be inherently ambiguous and therefore can have more than one syntax trees, such as *la belle porte le voile* where *belle, porte, voile* can be both noun/verb/noun or adjective/noun/verb. In addition, complex linguistic issues arise in the processing of utterances, where omission of a word or of a group of words—which otherwise are necessary for the grammatical completeness of a sentence—is tolerated. These issues often result in incorrect identification of verb arguments. For example, in *Henri veut aller au parc et sa mère à la bibliothèque* (Henri wants to go to the park and his mother to the library), the absence of a verb between words *mère* and *à la bibliothèque* may result in obtaining incorrect lexical categories for the remaining words.

VI. CONCLUSION AND FURTHER WORK

We have proposed an empirical approach towards building a semantic representation application for the French language, based on the CCG framework (for analyzing in the scope of the sentence) and on DRS (for dealing with semantic relations between the sentences of a discourse). Syntactic information and dependency relations between words are analyzed and extracted using a dependency parser. After that, the information is used to build a CCG derivation tree for each sentence. Finally, sentences in the form of CCG derivation trees are treated by Boxer, which we have adapted to French language by direct intervention on the source code, and we obtain a semantic representation of the discourse in FOL or in boxed format.

In future research, we plan to build a corpus with discourses and their meaning representation in DRS form, using this application. We also plan to use deep neural network models to improve the robustness of the results obtained.

ACKNOWLEDGEMENTS

This work has been supported by Crédit Mutuel Arkéa Bank. We also wish to thank the anonymous reviewers for their helpful comments.

REFERENCES

- [1] H. Kamp and U. Reyle, *From discourse to logic: Introduction to model-theoretic semantics of natural language, formal logic and discourse representation theory*. Springer, 1993.
- [2] M. Steedman, *Surface structure and interpretation*. MIT press, 1996.
- [3] J. Bos, “Wide-coverage semantic analysis with boxer,” in *Proceedings of the 2008 Conference on Semantics in Text Processing*. Association for Computational Linguistics, 2008, pp. 277–286.
- [4] J. M. Zelle and R. J. Mooney, “Learning to parse database queries using inductive logic programming,” in *Proceedings of the National Conference on Artificial Intelligence*, 1996, pp. 1050–1055.

- [5] F. Pereira and S. M. Shieber, *Prolog and natural-language analysis*. Microtome Publishing, 2002.
- [6] J. Bos, “Open-domain semantic parsing with Boxer,” in *Proceedings of the 20th nordic conference of computational linguistics*, 2015, pp. 301–304.
- [7] R. van Noord, L. Abzianidze, A. Toral, and J. Bos, “Exploring neural methods for parsing discourse representation structures,” 2018, preprint [arXiv:1810.12579](https://arxiv.org/abs/1810.12579).
- [8] J. Liu, S. B. Cohen, and M. Lapata, “Discourse representation structure parsing,” in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, vol. 1, 2018, pp. 429–439.
- [9] J. R. Curran, S. Clark, and J. Bos, “Linguistically motivated large-scale NLP with C&C and Boxer,” in *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*. Association for Computational Linguistics, 2007, pp. 33–36.
- [10] S. Clark and J. R. Curran, “Wide-coverage efficient statistical parsing with CCG and log-linear models,” *Computational Linguistics*, vol. 33, no. 4, pp. 493–552, 2007.
- [11] M. Lewis and M. Steedman, “A* CCG parsing with a supertag-factored model,” in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 990–1000.
- [12] R. Moot and C. G. Logics, “Grail: An automated proof assistant for categorial grammar logics,” 1998, <https://www.labri.fr/perso/moot/uitp.pdf>.
- [13] R. Moot, “Grail: an interactive parser for categorial grammars,” in *Proceedings of VEXTAL*, vol. 1999, 1999, pp. 255–261.
- [14] —, “Wide-coverage french syntax and semantics using Grail.”
- [15] A. Lefeuve, R. Moot, and C. Retoré, “Traitement automatique sur corpus de récits de voyages pyrénéens: une analyse syntaxique, sémantique et temporelle dans le cadre de la théorie des types,” in *3e Congrès Mondial de Linguistique française*, 2012, pp. 2485–2497.
- [16] M. Moortgat, “Categorial type logics,” in *Handbook of logic and language*. Elsevier, 1997, pp. 93–177.
- [17] G. V. Morrill, *Type logical grammar: Categorial logic of signs*. Springer, 2012.
- [18] M. Steedman and J. Baldridge, “Combinatory categorial grammar,” in *Non-Transformational Syntax: Formal and explicit models of grammar*. Wiley-Blackwell, 2011, pp. 181–224.
- [19] L. Dong and M. Lapata, “Language to logical form with neural attention,” 2016, preprint [arXiv:1601.01280](https://arxiv.org/abs/1601.01280).
- [20] C. Xiao, M. Dymetman, and C. Gardent, “Sequence-based structured prediction for semantic parsing,” in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, vol. 1, 2016, pp. 1341–1350.
- [21] R. Jia and P. Liang, “Data recombination for neural semantic parsing,” 2016, preprint [arXiv:1606.03622](https://arxiv.org/abs/1606.03622).
- [22] T. Kočiský, G. Melis, E. Grefenstette, C. Dyer, W. Ling, P. Blunsom, and K. M. Hermann, “Semantic parsing with semi-supervised sequential autoencoders,” 2016, preprint [arXiv:1609.09315](https://arxiv.org/abs/1609.09315).
- [23] J. Bos, V. Basile, K. Evang, N. Venhuizen, and J. Bjerva, “The Groningen Meaning Bank,” in *Handbook of Linguistic Annotation*, N. Ide and J. Pustejovsky, Eds. Springer, 2017, vol. 2, pp. 463–496.
- [24] L. Abzianidze, J. Bjerva, K. Evang, H. Haagsma, R. Van Noord, P. Ludmann, D.-D. Nguyen, and J. Bos, “The parallel meaning bank: Towards a multilingual corpus of translations annotated with compositional meaning representations,” 2017, preprint [arXiv:1702.03964](https://arxiv.org/abs/1702.03964).
- [25] M. Steedman, *The syntactic process*. MIT press, 2000.
- [26] K. Ajdukiewicz, “Die syntaktische Konnexität,” *Studia Philosophica*, vol. 1, pp. 1–27, 1935.
- [27] Y. Bar-Hillel, “A quasi-arithmetical notation for syntactic description,” *Language*, vol. 29, pp. 47–58, 1953.
- [28] H. B. Curry, R. Feys, W. Craig, R. Hindley, and J. P. Seldin, *Combinatory logic*. North-Holland Amsterdam, 1958.
- [29] P. Blackburn, J. Bos, M. Kohlhasse, and H. De Nivelle, “Inference and computational semantics,” in *Third International Workshop on Computational Semantics (IWCS-3)*, 1998.
- [30] J. Bos, “Computational semantics in discourse: Underspecification, resolution, and inference,” *Journal of Logic, Language and Information*, vol. 13, no. 2, pp. 139–157, 2004.
- [31] M. Candito, B. Crabbé, and P. Denis, “Statistical French dependency parsing: Treebank conversion and first results,” in *Seventh International Conference on Language Resources and Evaluation, LREC 2010*. European Language Resources Association (ELRA), 2010, pp. 1840–1847.
- [32] S. Green, M.-C. De Marneffe, J. Bauer, and C. D. Manning, “Multiword expression identification with tree substitution grammars: A parsing tour de force with French,” in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2011, pp. 725–735.
- [33] B. Guillaume and G. Perrier, “Dependency parsing with graph rewriting,” in *Proceedings of the 14th International Conference on Parsing Technologies*, 2015, pp. 30–39.
- [34] N. L. Lê and Y. Haralambous, “CCG Supertagging Using Morphological and Dependency Syntax Information,” in *Proceedings of the 20th International Conference on Computational Linguistics and Intelligent Text Processing (CICLING)*, 2019.
- [35] K. K. Schuler, “VerbNet: A broad-coverage, comprehensive verb lexicon,” Ph.D. dissertation, University of Pennsylvania, 2005.
- [36] Q. Pradet, L. Danlos, and G. De Chalendar, “Adapting VerbNet to French using existing resources,” in *LREC’14, Ninth International Conference on Language Resources and Evaluation*, 2014.
- [37] J. Bos, “Implementing the binding and accommodation theory for anaphora resolution and presupposition projection,” *Computational Linguistics*, vol. 29, no. 2, pp. 179–210, 2003.
- [38] A. Abeillé, L. Clément, and F. Toussenet, “Building a treebank for french,” in *Treebanks*. Springer, 2003, pp. 165–187.