



# Deeper spoken language understanding for man-machine dialogue on broader application domains: a logical alternative to concept spotting.

Jeanne Villaneau, Jean-Yves Antoine

## ► To cite this version:

Jeanne Villaneau, Jean-Yves Antoine. Deeper spoken language understanding for man-machine dialogue on broader application domains: a logical alternative to concept spotting.. EACL 2009 WORKSHOP ON SEMANTIC REPRESENTATION OF SPOKEN LANGUAGE, Mar 2009, Athènes, Greece. pp.50-57. hal-00515885

**HAL Id: hal-00515885**

**<https://hal.science/hal-00515885>**

Submitted on 8 Sep 2010

**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.

# Deeper spoken language understanding for man-machine dialogue on broader application domains: a logical alternative to concept spotting

Jeanne Villaneau

UEB (Université Européenne de Bretagne) Université François Rabelais - Tours

VALORIA

France

villaneau@univ-ubs.fr

Jean-Yves Antoine

Université François Rabelais - Tours

LI

France

Jean-Yves.Antoine@univ-tours.fr

## Abstract

LOGUS is a French-speaking spoken language understanding (SLU) system which carries out a deeper analysis than those achieved by standard concept spotters. It is designed for multi-domain conversational systems or for systems that are working on complex application domains. Based on a logical approach, the system adapts the ideas of incremental robust parsing to the issue of SLU. The paper provides a detailed description of the system as well as results from two evaluation campaigns that concerned all of current French-speaking SLU systems. The observed error rates suggest that our logical approach can stand comparison with concept spotters on restricted application domains, but also that its behaviour is promising for larger domains. The question of the generality of the approach is precisely addressed by our current investigations on a new task: SLU for an emotional robot companion for young hospital patients.

## 1 Introduction

Despite the indisputable advances of automatic speech recognition (ASR), highly spontaneous speech remains an important barrier to the wide spreading of speech based applications. The goal of spontaneous speech understanding remains feasible, provided the interaction between the user and the system is restricted to a task-oriented dialogue (restricted vocabulary). Present research is investigating mixed or user initiated dialogue for less restricted tasks. It is the purpose of this paper, which focuses on spontaneous speech understanding in such complex applications.

Generally speaking, information speech dialogue systems are based on the same architecture.

At first, a speech recognizer processes the speech signal and provides a string (or a lattice) of words that should correspond to the spoken sentence. Then, this string is parsed by a spoken language understanding module (SLU) in order to build a semantic representation that represents its propositional meaning. Finally, this semantic structure is sent to a dialogue manager which controls the interaction with the user (database interrogation, dialogue management, answer generation). The answers to the user can be displayed on screen and/or through a message generated by a text-to-speech synthesis. This paper focuses on the SLU module of such a dialogue system. On the whole, SLU has to cope with two main difficulties:

- speech recognition errors: highly spontaneous speech remains hard to recognize for current ASR systems (Zue et al., 2000). Therefore, the SLU module has to work on a strongly corrupted string of words.
- spoken disfluencies: filled pauses, repetitions and repairs make the parsing of conversational spoken language significantly harder to achieve (Heeman, Allen, 2001).

In order to overcome those difficulties, most SLU systems follow a selective strategy which comes down to a simple concept spotting: they restrict the semantic analysis to a mapping of the sentence with the main expectations of the user in relation with the task (Minker W. et al., 1999; Bangalore S. et al., 2006). Consider, for instance, an air transport information system and the following spoken utterance:

(1) *Cou- could you list me the flights uh the scheduled flights for Tenerife Tenerife Tenerife North please*

Satisfying the speaker's goals only requires detecting the nature of their requests (list flights) and the required destination (Tenerife North). Those

two concepts (list, Tenerife North) will fill a shallow semantic frame which is supposed to represent the useful meaning of the sentence. Such task-driven approaches meet, to a great extent, the needs of SLU in terms of robustness, since they only involve a partial analysis of the sentence. Whether the processing is based on a statistical or a knowledge-based approach, several evaluation campaigns proved that concept spotting is suitable for spoken language understanding, provided the application task is sufficiently restricted. However, concept spotters suffer from noticeable limitations:

- Although they resist gracefully speech recognition errors, they are not able to detect their eventual presence, since they do not consider the global structure of the sentence. This limitation can be particularly penalizing when the error is related to a key element, for example when the error prevents the system to determine the type (dialogue act) of the utterance. Indeed, concept spotters often base SLU on the initial characterization of the question type. When analyzing the errors of his statistical concept spotter, Minker has shown that the correct identification of the question type is a key issue in terms of final robustness (Minker W. et al., 1999).
- Since they are based on the identification of rather flat semantic frames, these approaches hardly succeed in representing complex syntactic relations such as overlapping coordinate phrases or negations.
- Although it is well known that generality is an important issue for SLU, this question is generally approached in term of technical portability from one (narrow) task to another. Now, one should wonder whether concept spotting is still suitable on larger application domains. It seems that the robustness of the spotting process depends strongly on the degree of lexical ambiguity of the considered task. For instance, Bousquet has shown that the concept error rate of her stochastic spotter is two times higher on ambiguous words than on non ambiguous ones (Bousquet et al., 2003).

Such considerations tend to show that to apply concept spotting to more complex tasks could be

difficult. Such observations are well known (Zechner K., 1998; Van Noord et al., 1999), and noticeable attempts have already been done to reach a deeper semantic analysis. However, statistical or knowledge-based concept spotting remains the prevailing paradigm in SLU, mainly because of engineering motivations (quick and easy building). On the contrary, we have decided to develop a SLU system (LOGUS<sup>1</sup>) which carries out a complete analysis of the utterance while keeping the robustness of standard concept spotting approaches. The system, which is based on a logical approach, adapts the ideas of incremental robust parsing (Aït-Mokhtar S., 2002; Basili, 2003) to the issue of speech conversational systems. In section 2, we will describe the system into detail. Then, section 3 will present results from different evaluation campaigns in which we participated. These experiments concerned standard restricted tasks (hotel reservation for instance) for which concept spotting is well adapted. As a result, this section does not aim to prove a superiority of our approach, but simply to show that this deeper processing is able to keep a satisfactory robustness, by comparison with prevailing approaches. Finally, we give in section 4 a brief description on our present work concerning the integration of LOGUS in a conversational robot which is dedicated to general interaction with children who are in hospital for a long-stay. This example will illustrate the portability abilities of our approach on complex application tasks, in addition with our previous works on general tourism information.

## 2 Description of the LOGUS system

The task of a SLU is to turn a sequence or a graph of words into a semantic representation; so a SLU system has to perform a translation from natural language to a formal target language. This section begins with the description of the formal language chosen for the LOGUS system. We then explain the basic principles of parsing and its main steps.

### 2.1 Semantic representation

When it comes to the choice of a target language for the system, the following points must be taken into account.

- We want to implement automatic understanding in application domains where predefined

---

<sup>1</sup>LOGical Understanding System.

semantic frames are not sufficient to represent all the possible queries (Van Noord et al., 1999). Furthermore, any SLU aims at providing results usable by a dialogue manager: the target language must reconcile simplicity with precision.

- This semantic representation must obviously extend to a pragmatic one. That means that it should involve the characterization of the dialogue acts related to the speech turn (Austin J.-L., 1962).

We have chosen a formalism compatible with these constraints and inspired by the illocutionary logic of D. Vanderveken (Vanderveken D., 1981). In this formalism, the form of an elementary illocutionary act is  $F(P)$  where  $F$  is the illocutionary force, and  $P$  its propositional content.

The LOGUS system thus provides a logical formula as the semantic representation of an utterance. A *language act* contains clues about the intentions of the speaker: it is labelled illocutionary force, while the propositional content is a structure built with the domain objects and their properties which is called an *object string*.

The following example shows a single speech turn uttered for a tourism information system:

(2) *j'ai réservé une chambre dans un deux étoiles l'hôtel euh l'hôtel Rex pour y aller d'ici comment est-ce que je peux faire (I booked a room in a two-star hotel in the hotel hum in the Rex hotel from here how can I go at there)*

This turn expresses two different language acts, which is quite usual in conversational speech: a piece of information (*I booked a room...*) is followed by the user question (*... how can I go....*). Such complex speech turns are difficult to analyze for concept spotters, since they usually base the parsing on one language act detection. The logical formula LOGUS provides is split into two language acts: (*information act*) and (*question how*). The second act is interpreted by the system in the context of the first one:

```
((information act)
  (of (reservation [])
    (hotel [(ident. (name "Rex")), (star (int 2))])))
(question how)
  (to-go [(to (contextual_location []),
    (from (hotel [(ident. (name "Rex"))]))))])
```

In the formula, *reservation*, *hotel* and *to-go* are object labels; (*ident. (name "Rex")*), (*star (int 2)*)

are properties. The two objects of labels *reservation* and *hotel* are linked with the generic relation *of*, which indicates a subordination relation. It is the main relation, (in addition with logical coordinations *and*, *or* and *not*) which is used for building complex object strings.

## 2.2 General system architecture

Incremental parsing methodology is used for text parsing in order to combine efficiency with robustness (Aït-Mokhtar S., 2002). With LOGUS, we tried to show that such methods can be extended to spoken language parsing.

The system has to parse out-of-grammar constructions but spoken language studies have shown that minimal syntactic structures are generally preserved in repairs and false-starts (Mc Kelvie D., 1998). We have thus chosen to carry out an incremental bottom-up parsing, where words are gradually combined. At the beginning, the parser groups words according to mainly syntactic rules in order to form minimal chunks that correspond to basic concepts of the application domain. Then, as word group size increases, their meaning becomes more precise, enough to relax syntactic criteria and thereby overcome the problem of out-of-grammar sentences.

The general architecture of the system is shown in Figure 1. The parsing is essentially split into three stages. The first stage is *chunking* (Abney S., 1991) where grammatical words are linked to the lexical words to which they are referred. The following stage gradually builds links between the chunks in order to detect semantic relations between the corresponding concepts, and the last one achieves a contextual interpretation (anaphoric resolution for instance). The process of building links between chunks and contextual understanding uses a domain ontology.

Only one formalism is used during these parsing stages. It is designed to distinguish syntax and semantics and to preserve genericity of the parsing rules. Each component is specified by a list of what we can call definitions; each of them is a triplet  $\langle C, R, T \rangle$  where

$C$ : is a syntactic label, called *syntactic category*: for example *adjective*, (*verb I present*).

$R$ : points out the semantic function of the component. It is called *semantic role*: for example *object*, (*prop price*) where *prop* is for property.

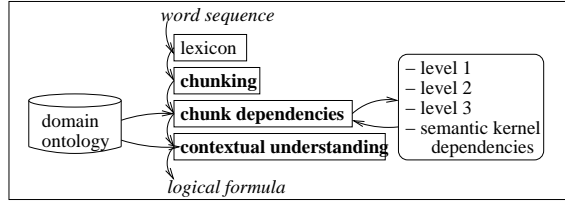


Figure 1: General architecture of the LOGUS system

$T$ : is the *semantic translation*. It is an element of the logical formula built by the system. It belongs thus to the target language.

The first two triplet elements,  $C$  and  $R$ , are widely domain independent. A basic principle is to define parsing rules from these elements in order to preserve the genericity of the system. Each parsing rule combines two or three triplets in order to build a new result triplet.

### 2.3 Chunking

Our experiments with LOGUS have clearly shown that *chunking* is effective for spoken language, provided the chunks are very short: more precisely, errors made at the speech recognition level make it dangerous to link objects or properties according to pure syntactic criteria, without checking these links with semantic criteria. Therefore the *chunks* built by LOGUS include only one content word: we call them *minimal chunks*. *Chunking* is based on the principle of linking function words to the near content word.

The formalism used in this step is inspired by Categorical Grammars of the AB type<sup>2</sup>, whose rules are generalized from the first two elements of the constituent triplets. Function words have definitions in which syntactic category and semantic role are fractional. In such definitions, the semantic translation is a  $\lambda$ -abstraction (in the  $\lambda$ -calculus meaning)<sup>3</sup>. The semantic translation of the result triplet is achieved by applying this abstraction to the semantic translation of the un-fractional triplet. Formally, the following two rules are applied, where  $F$  is an abstraction:

$$\begin{aligned} & \langle C_A/C_B, R_A/R_B, F \rangle, \langle C_B, R_B, S_B \rangle \\ & \quad \rightarrow \langle C_A, R_A, (F S_B) \rangle \\ & \langle C_B, R_B, S_B \rangle, \langle C_B \setminus C_A, R_B \setminus R_A, F \rangle \\ & \quad \rightarrow \langle C_A, R_A, (F S_B) \rangle \end{aligned}$$

<sup>2</sup>The formalism can be expressed in terms of pregroup formalism too (Lambek J., 1999).

<sup>3</sup>LOGUS is implemented in  $\lambda$ Prolog, a logic programming language whose terms are  $\lambda$ -terms with simple types.

In the following example only one definition is shown for each component ( $gn$  is for *nominal group*).

	<i>trois (three)</i>	<i>étoiles (stars)</i>
C	adj_num	adj_num \ gn
R	(prop nb)	(prop nb) \ (prop nb_star)
S	(int 3)	$\lambda x. (star x)$

By applying the second rule, we obtain the following chunk:

$$\begin{aligned} & \text{"trois étoiles" (three stars)} \\ & \langle gn, (prop nb\_star), (star (int 3)) \rangle. \end{aligned}$$

The semantic translation of the result triplet is obtained by  $\beta$ -reduction of the  $\lambda$ -term  $(\lambda x. (star x) (int 3))$ . For example, the utterance

(3) "À l'hôtel Caumartin quels sont le les tarifs pour pour une chambre double" (In Caumartin hotel what are the the prices for for a double room) is segmented into six chunks during the *chunking* stage. Their semantic translations are:

- [1] (*hotel []*),
- [2] (*identity (name "Caumartin") []*),
- [3] (*what (interrogation)*), [4] (*price []*),
- [5] (*room []*), [6] (*size double*).

At the end of the *chunking* process, the determiner *le* and the first occurrence of the preposition *pour* are deleted because they are fragments without semantic content. Deletions such as these are a first way of dealing with repairs.

### 2.4 Domain ontology

The limited scope of the application domain makes it possible to describe exhaustively the pragmatic and semantic domain knowledge. A domain ontology specifies how objects and properties can be compounded. The handled processings are expected to be generic while using a domain dependent ontology: to achieve that, the ontology is defined by generic predicates whose domain objects and domain properties are the arguments.

For example, the possibility of building the conceptual relation *of* between two objects (cf. 2.1) is defined by the predicate *is\_sub\_object* whose arguments are two object labels: so the relation *is\_sub\_object(room, hotel)* expresses a part-whole relation possibility between such two objects.

### 2.5 Chunk dependencies

Chunk dependencies are built by an incremental process which is compound of several successive stages. Each stage is based on rewriting rules

which are specified from the first two components of the constituent triplets and from the generic ontology predicates. They are thus not specific to the domain of application, what assures, to a certain extent, the genericity of the process.

Consider for instance the following rule, which leads to the binding of two consecutive chunks which share a meronomic (part\_of) relation:

$$\frac{\begin{array}{l} \langle C_1, \text{object}, O_1 \rangle, \langle C_2, \text{object}, O_2 \rangle \\ - O_1 \text{ simple object of label } Et_1 \\ - O_2 \text{ object string of label } Et_2 \\ - \text{is\_sub\_object}(Et_1, Et_2) \end{array}}{\langle C, \text{object}, (\text{of } O_1 O_2) \rangle}$$

where  $C$  is obtained by composing  $C_1$  and  $C_2$ .

As an illustration, this rule will form a complex object (*of (price []) (room [(size double)])*) from the initial two chunks (*price []*) and (*room (size double)*). This rule is completely generic and should apply on any task. The knowledge specific to the task intervenes only on the definition of the predicate *is\_sub\_object*. As a result, one could speak of procedural genericity to qualify our system.

As long as possible, the first processing stages try to respect syntactic criteria. However, in presence of spoken disfluencies or speech recognition errors, it is likely that the utterance is out-of-grammar. Therefore, since the detected links between chunks make the meaning of the linked chunks more specific, the next stage tries to detect chunk dependencies more on more on semantic or pragmatic features only. Subsequently, studying dependencies between the components makes it possible to eliminate some components, especially in the case of word recognition errors.

As an illustration, Figure 2 shows how links are gradually built during the parsing stage of utterance (3) (cf. section 2.3). The chunks are in rectangular boxes in dotted lines.

The first step of chunk binding links the first two chunks into the object:

*(hotel [(ident. (name "Caumartin"))])*.

The second step links the object (*room []*) with the property (*size (double)*) to obtain the object (*room [(size double)]*). Then, the two objects *price* and *room* are linked with the conceptual relation *of* to obtain (*of (price []) (room [(size double)])*) and this object string is connected to the language act: (*question what*). The position of the prepositional phrase *à l'hôtel Caumartin* is not usual in French syntagmatic ordering. It is indeed an example of

extraposition which is not accepted by the syntactic constraints considered by the system. As a result, the conceptual relation *of*, which links the object of label *room* with the object (*hotel [(ident. (name "Caumartin"))]*) is built later, when these constraints are relaxed.

## 2.6 Contextual understanding

Many sentences are elliptical and incomplete in a dialogue. Therefore, it is necessary to use the current context of the task and the dialogue history in order to complete their understanding. The objectives of the contextual understanding in LOGUS are thus close to the objectives of the authors of the OntoSem system (McShane M., 2005): the completion of semantic fragments. Reference resolution is thereby extended to a more general completion of the semantic representation.

While syntactic anaphora criteria are generally respected in texts, anaphora gender and number are frequently broken in spoken language. Moreover, gender and number morphological marks are hardly perceptible in spoken French. They are therefore very often corrupted by speech recognition errors. So, in the LOGUS system, anaphora resolution is based on the same principles as the rest of the parsing: combining syntactic and semantic criteria. Both nominal and pronominal anaphora (with definite expressions) are considered during this contextual interpretation stage.

Completion is based on the concept of *object string*. A property or an object may be completed by an "over-object" of the context, if the ontology makes it possible to do so. For example, the object *price* of the sentence "*quel est le tarif*" (*what is the price*) is automatically completed in (*of (price []) (of (room []) (hotel [(name "Rex"))])*) if the *object string* (*of (room []) (hotel [(name "Rex"))])* is an *object string* which is part of the previous utterance.

## 3 Evaluations and results

LOGUS is a French-speaking system. It took part in the two evaluation campaigns that were carried out in the last year designed for French spoken language understanding: the GDR-I3 challenge-based campaign and the MEDIA project.

### 3.1 The GDR-I3 campaign

LOGUS took part in the challenge-based campaign, held by the GDR-I3 consortium of the

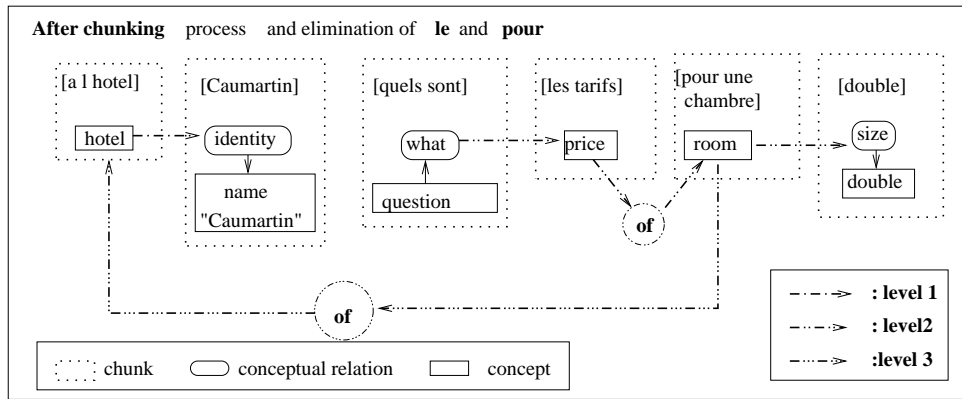


Figure 2: Characterization of chunk dependencies : example on the utterance “à l’hôtel Caumartin quels sont les tarifs pour une chambre double” (in Caumartin hotel what are the the prices for for a double room).

French CNRS research agency (Antoine et al., 2002). We won’t describe here in detail the results of this campaign, since it concerned a former version of LOGUS. It seems however interesting to analyse the distribution of the errors made by LOGUS to have an idea of the benefits of our approach. The evaluation corpus was divided among several tests which were respectively related to a specific difficulty: speech recognition errors, speech repairs and other disfluences, and finally messages of a structural complexity (embedded coordination or subordination, for instance) significantly higher than those usually met in standard ATIS-like application domains.

The distribution of the concept error rates of the LOGUS SLU system is the following:

Speech recognition:	9.5%
Complex structures:	9.8%
Repairs:	15%

It should be noted here that the robustness of LOGUS decreases rather gracefully on complex messages, while SLU systems based on concept spotting meet real difficulties on such utterances. For instance, Cacao (Bousquet-Vernhettes et al., 1999; Bousquet-Vernhettes et al., 2003) is a concept spotter which participated to the GDR-I3 campaign. It has been shown that most of its errors resulted from its difficulties to resolve lexical ambiguities in complex sentences. This observation suggests that our logical deep parsing should fulfill better than concept spotting the needs of complex application domains such as general purpose tourist information or collaborative planning (Allen J. et al., 2002), or even multi-domain applications (Dzikovska M. et al., 2005). Unfortun-

nately, French evaluation campaigns have never investigated such difficult tasks.

### 3.2 The MEDIA project

MEDIA-EVALDA was an evaluation campaign hold by the French Ministry of Research. It concerned all the French laboratory working on SLU. Once again, this evaluation investigated a rather restricted application domain: hotel reservation. It is well known that concept spotters fit successfully such simple tasks. Nevertheless, we decided to take part in this evaluation in order to see to which extent LOGUS should be compared to standard concept spotters in such disadvantageous conditions.

Participants defined reservation scenarios which were used to build a corpus made up of 1250 recorded dialogues. Recording used a WOZ system simulating vocal tourist phone server (Devillers et al., 2004). The MEDIA corpus, which is made up of real-life French spontaneous dialogues, is surely to become a benchmark reference for French contextual SLU.

The evaluation paradigm forced every participant to convert his own semantic representation into a common reference, which relies-on an attribute/value frame: each utterance is divided into semantic segments, aligned on the sentence, and each segment is represented by a triplet: (mode, attribute, value). Relations between attributes are represented by their order in the representation and the composed attribute names.

Nine systems participated to this first campaign. An error was count for any difference with one of the elements of the reference (mode, attribute

System	1	2	3	4 (LOGUS)	5
Approach	concept spotting	concept spotting	syntactic deep parsing	logical deep parsing	concept spotting
Error rate	29.0%	30.3%	36.3%	37.8%	41.3%

Table 1: MEDIA results.

or value). Table 1 summarises the results of the best five systems. At first glance, one should find the reported error rates rather deceptive. However, one must realize that the test corpus involved highly spontaneous conversational speech, with very frequent speech disfluences. As a result, these results should be compared, for instance, to ASR errors rates observed on the SWITCHBOARD corpus (Greenberg S. et al., 2000).

LOGUS was ranked fourth and its robustness was rather close to the best participants. Now, if you consider that the systems ranked 1st, 2nd and 5th were using a concept spotter, these results shows that our approach can bear comparison with standard approaches even on this task. These encouraging performances suggest that it is possible to achieve a deep understanding of conversational speech while respecting at the same time some robustness requirements: our approach seems indeed competitive even in a domain where concept spotters are known to be very efficient. To our mind, the interest of our approach is that this robustness should remain on larger application domains. We are precisely trying to test this genericity by adapting LOGUS to a wider application domain in the framework of the Emotirob project.

#### 4 Genericity and portability experiment

We are currently testing the portability of our approach by adapting LOGUS to a really different task, which corresponds to an unrestricted application domain, general purpose understanding of child language, with additional emotional state detection. The whole project, supported by ANR (National French Research Agency), aims at achieving a robot companion which can interact with sick or disabled young children with the help of facial expressions. Although the robot does not have to react to every speech act of the child, we have to deal with spoken understanding in an unrestricted domain. Fortunately, the age of the children involved (3-5) implies a restricted vocabulary. This work is still in progress. Our first investigations suggest however that LOGUS is a

suitable understanding system for the pursued purpose: since there will never be significant corpora related to this kind of task, we can't use statistical methods. Moreover, because of the genericity of LOGUS, the main part of the analysis can be reused without important changes. Thus, three-month work was enough to build a first prototype of the system and the problem is restricted to the main problem of this project: building an ontology which models the cognitive and emotional world of young children.

The generality of the used formalism makes it possible to include an emotional component by turning the triplet structure into a quadruplet structure. Of course, composition rules have to include this new component. We are currently working on the computation of the emotional states from both prosodic and lexical cues. Whereas many works have investigated a prosodic-based detection (Devillers et al., 2005), word-based approaches remain quite original. Our hypothesis is that emotion is compositional, e.g. that is possible to compute the global emotion carried by a sentence from the emotion of every content word. This calculation depends obviously of the semantic structure of the utterance: our system will precisely benefit from the characterization of the chunk dependencies carried on by LOGUS. For the moment being, we are working on the definition of a complete lexical norm of emotional values from children of 3, 5 and 7 years. This norm will be established in collaboration with psycholinguists from Montpellier University, France.

#### 5 Conclusion

When we started implementing the LOGUS system, one of our objectives was to achieve robust parsing of spontaneous spoken language while making the application domain much wider than is currently done. Logical formalisms are not usually viewed as efficient tools for pragmatic applications. The promising results of LOGUS show that they can be brought into interesting new approaches.



Another objective was to have a rather generic system, despite the use of a domain-based semantic knowledge. We have fulfilled this constraint through the definition of generic predicates as well as generic rules working on semantic triplets or quadruplets which makes it possible to have generic chunk linking rules. The performances of LOGUS show that a deeper understanding can bear comparison with concept spotting approaches.

## References

- Abney S. 1991. Parsing by Chunks. *Principle Based Parsing*. R. Berwick, S. Abney and C. Tenny Eds. Kluwer Academix Publishers.
- Aït-Mokhtar S., Chanod J.-P. and Roux C. 2002. Robustness beyond Shallowness: Incremental Deep Parsing. *Natural Language Engineering*, 8 (2-3): p. 121–144.
- Allen J. and Ferguson G. 2002. Human-Machine Collaborative Planning. *Proc. of the 3rd International NASA Workshop on Planning and Scheduling for Space*, Houston, TX.
- Antoine J.-Y. et al. 2002. Predictive and Objective Evaluation of Speech Understanding: the "challenge" evaluation campaign of the I3 speech workgroup of the french CNRS. *Proceedings of the LREC 2002, 3<sup>rd</sup> International Conference on Language Resources and Evaluation*, Las Palmas, Spain.
- Austin J.-L. 1962. *How to do things with words*. Oxford.
- Bangalore S., Hakkani-Tür D. and Tür G. 2006. *Special issue on Spoken Language Understanding in Conversational Systems*. Speech Communication. 48.
- Basili R. and Zanzotto F.M. 2003. Parsing engineering and empirical robustness. *Natural Language Engineering*. 8 (2-3).
- Bousquet-Vernhettes C., Bouraoui J.-L. and Vigouroux N. 2003. Language Model Study for Speech Understanding. *Proc. International Workshop on Speech and Computer (SPECOM'2003)*, Moscow, Russia, p. 205–208.
- Bousquet-Vernhettes C., Privat R. and Vigouroux N. 2003. Error handling in spoken dialogue systems: toward corrective dialogue. *ISCA workshop on Error Handling in Spoken Dialogue Systems*, Chteau-d'Oex-Vaud, Suisse, p. 41–45.
- Bousquet-Vernhettes C., Vigouroux N. and Pérennou G. 1999. Stochastic Conceptual Model for Spoken Language Understanding. *Proc. International Workshop on Speech and Computer (SPECOM'99)*, Moscow, Russia, p. 71–74.
- Devillers L. et al. 2004. The French Evalda-Media project: the evaluation of the understanding capabilities of Spoken Language Dialogue Systems. *Proceedings of the LREC 2004, 4<sup>th</sup> International Conference on Language Resources and Evaluation*, Lisboa, Portugal.
- Devillers L., Vidrascu, L. and Lamel, L. 2005. Challenges in real-life emotion annotation and machine learning based detection. *Neural Networks*, 18, p. 407–422.
- Dzikovska M., Swift M. and Allen J. and de Beaumont W. 2005. Generic parsing for multi-domain semantic interpretation. *Proc. 9th International Workshop on Parsing Technologies (IWPT05)*, Vancouver BC.
- Greenberg S. and Chang, S. 2000. Linguistic dissection of switchboard-corpus automatic speech recognition systems. *Proc. ISCA Workshop on Automatic Speech Recognition: Challenges for the New Millennium*, Paris, France.
- Heeman P. and Allen J. 2001. Improving robustness by modeling spontaneous events. *Robustness in language and speech technology*, Kluwer Academics. Dordrecht, NL. p. 123–152.
- Lambek J. 1999. Type grammars revisited. *Logical Aspects of Computational Linguistics*, A. Lecomte, F. Lamarche and G. Perrier (eds), LNAI 1582, Springer, Berlin, p. 1–27.
- Mc Kelvie D. 1998. The syntax of disfluency in spontaneous spoken language. *HCRC Research Paper*, HCRC/RP-95.
- McShane M. 2005. Semantics-based resolution of fragments and underspecified structures. *Traitement Automatique des Langues*, 46(1): p. 163–184.
- Minker W., Waibel A. and Mariani J. 1999. *Stochastically based semantic analysis*. Kluwer Ac., Amsterdam, The Netherlands.
- Vanderveken D. 2001. Universal Grammar and Speech act Theory. *Essays in Speech Act Theory*. Eds J. Benjamin, D. Vanderveken and S. Kubo, p. 25–62.
- van Noord G., Bouma G. and Koeling R. and Nederhof M. 1999. Robust grammatical analysis for spoken dialogue systems. *Natural Language Engineering*. 5(1): p. 45–93.
- Zechner K. 1998. Automatic construction of frame representations for spontaneous speech in unrestricted domains. *COLING-ACL'1998. Montreal, Canada*. p. 1448–1452.
- Zue V., Seneff S., Glass J., Polifrini J., Pao C., Hazen T.J. and Hetherington L. 2000. Jupiter: a telephone-based conversational interface for weather information. *IEEE Transactions on speech and audio processing*. 8(1).