Terminology Finite-State Preprocessing for Computational LFG

Caroline Brun
Xerox Research Centre Europe
6, chemin de Maupertuis 38240 Meylan France
Caroline.Brun@xrce.xerox.com

Abstract

This paper presents a technique to deal with multiword nominal terminology in a computational Lexical Functional Grammar. This method treats multiword terms as single tokens by modifying the preprocessing stage of the grammar (tokenization and morphological analysis), which consists of a cascade of two-level finite-state automata (transducers). We present here how we build the transducers to take terminology into account. We tested the method by parsing a small corpus with and without this treatment of multiword terms. The number of parses and parsing time decrease without affecting the relevance of the results. Moreover, the method improves the perspicuity of the analyses.

1 Introduction

The general issue we are dealing with here is to determine whether there is an advantage to treating multiword expressions as single tokens, by recognizing them before parsing. Possible advantages are the reduction of ambiguity in the parse results, perspicuity in the structure of analyses, and reduction in parsing time. The possible disadvantage is the loss of valid analyses. There is probably no single answer to this issue, as there are many different kinds of multiword expressions. This work follows the integration[1] of (French) fixed multiword expressions like *a priori*, and time expressions, like *le 12 janvier 1988*, in the preprocessing stage.

Terminology is an interesting kind of multiword expressions because such expressions are almost but not completely fixed, and there is an intuition that you won't loose many good analyses by treating them as single tokens. Moreover, terminology can be semi or fully automatically extracted. Our goal in the present paper is to compare efficiency and syntactic coverage of a French LFG grammar on a technical text, with and without terminology recognition in the preprocessing stage. The preprocessing consists mainly in two stages: tokenization and morphological analysis. Both stages are performed by use of finite-state lexical transducers (Kartunnen, 1994). In the following, we describe the insertion of terminology in these finite-state transducers. as well as the consequences of such an insertion on the syntactic analysis, in terms of number of valid analyses produced, parsing time and nature of the results. We are part of a project. which aims at developing LFG grammars, (Bresnan and Kaplan, 1982), in parallel for French, English and German, (Butt et al., To appear). The grammar is developed in a computational environment called XLE (Xerox Linguistic Environment), (Maxwell and Kaplan, 1996), which provides automatic parsing and generation, as well as an interface to the preprocessing tools we are describing.

2 Terminology Extraction

The first stage of this work was to extract terminology from our corpus. This corpus is a small French technical text of 742 sentences (7000 words). As we have at our disposal parallel aligned English/French texts, we use the English translation to decide when a potential term is actually a term. The terminology we are dealing with is mainly nominal. To perform this extraction task, we use a tagger (Chanod and Tapanainen, 1995) to disambiguate the French text, and then extract the following syntactic patterns, *N Prep N*, *N N* , *N A*, *A N*, which are good candidates to be terms. These candidates

[1] This integration has been done by Frédérique Segond.

are considered as terms when the corresponding English translation is a unit, or when their translation differs from a word to word translation. For example, we extract the following terms:

(1) *vitesses rampantes (creepers)*
boite de vitesse (gearbox)
arbre de transmission (drive shaft)
tableau de bord (instrument panel)

This simple method allowed us to extract a set of 210 terms which are then integrated in the preprocessing stages of the parser, as we are going to explain in the following sections.
We are aware that this semi-automatic process works because of the small size of our corpus. A fully automatic method (Jacquemin, 1997) could be used to extract terminology. But the material extracted was sufficient to perform the experiment of comparison we had in mind.

3 Grammar Preprocessing

In this section, we present how tokenization and morphological analysis are handled in the system and then how we integrate terminology processing in these two stages.

3.1 Tokenization

The tokenization process consists of splitting an input string into tokens, (Grefenstette and Tapanainen, 1994), (Ait-Mokthar, 1997), i.e. determining the word boundaries. If there is one and only one output string the tokenization is said to be deterministic, if there is more than one output string, the tokenization is non deterministic. The tokenizer of our application is non deterministic (Chanod and Tapanainen, 1996), which is valuable for the treatment of some ambiguous input string[2], but in this paper we deal with fixed multiword expressions.
The tokenization is performed by applying a two-level finite-state transducer on the input string. For example, applying this transducer on the sentence in 2 gives the following result, the token boundary being the @ sign.

(2) Le tracteur est à l'arrêt.
(The tractor is stationary.)
Le@tracteur@est@à@l'@arrêt@.@

In this particular case, each word is a token. But several words can be a unit, for example compounds, or multiword expressions. Here are some examples of the desired tokenization, where terms are treated as units:

(3) La boîte de vitesse est en deux sections.
(the gearbox is in two sections)
La@boîte de vitesse@est@en@deux@sections@.@

(4) Ce levier engage l'arbre de transmission.
(This lever engages the drive shaft.)
Ce@levier@engage@l'@arbre de transmission@.@

We need such an analysis for the terminology extracted from the text. This tokenization is realized in two logical steps. The first step is performed by the basic transducer and splits the sentence in a sequence of single word. Then a second transducer containing a list of multiword expressions is applied. It recognizes these expressions and marks them as units. When more than one expression in the list matches the input, the longest matching expression is marked. We have included all the terms and their morphological variations in this last transducer, so that they are analyzed as single tokens later on in the process. The problem now is to associate a morphological analysis to these units.

3.2 Morphological Analysis

The morphological analyzer used during the parsing process, just after the tokenization process, is a two-level finite-state transducer (Chanod, 1994). This lexical transducer links the surface form of a string to its morphological analysis, i.e. its canonical form and some characterizing morphological tags. Some examples are given in 5.

(5) >veut
vouloir+IndP+SG+P3+Verb
>animaux
animal+Masc+PL+Noun
animal+Masc+PL+Adj

The compound terms have to be integrated into this transducer. This is done by developing a local regular grammar which describes the compound morphological variation, according to the inflectional model proposed in (Kartunnen et al., 1992).
The hypothesis is that only the two main parts

[2] for example *bien que* in French

of the compounds are able to vary. i.e. N1 or A1, and N2 or A2. in the patterns *N1 prep N2*, *N1 N2*, *A1 N2*, and *N1 A2*. In our corpus, we identify two kinds of morphological variations:

- The **first** part varies in **number**:
  gyrophare de toit, *gyrophares de toit*
  régime moteur, *régimes moteur*
- **Both** parts vary in **number**:
  roue motrice, *roues motrices*

This is of course not general for French compounds; there are other variation patterns, however it is reliable enough for the technical manual we are dealing with. Other inflectional schemes and exceptions are described in (Karttunen et al., 1992) and (Quint, 1997), and can be easily added to the regular grammar if needed.
A cascade of regular rules is applied on the different parts of the compound to build the morphological analyzer of the whole compound. For example, *roue motrice* is marked with the diacritic +DPL, for double plural and then, a first rule which just copies the morphological tags from the end to the middle is applied if the diacritic is present in the right context:

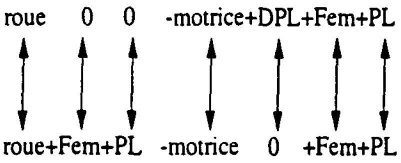

Figure 1: First rule

A second rule is applied to the output of the preceding one and "realizes" the tags on surface.

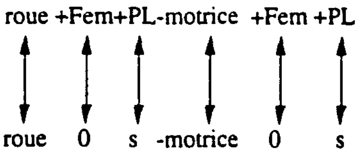

Figure 2: Second rule

The composition of these two layers gives us the direct mapping between surface inflected forms and morphological analysis. The same kind of rules are used when only the first part of the compound varies, but in this case the second rule just deletes the tags of the second word. The two morphological analyzers for the two variations are both unioned into the basic morphological analyzer for French we use for morphology. The result is the transducer we use following tokenization and completing input preprocessing. An example of compound analysis is given here:

(6) > roues motrices
roue motrice+Fem+PL+Noun
> régimes moteur
régime moteur+Masc+PL+Noun

The morphological analysis developed here for terminology allows multiword terms to be treated as regular nouns within the parsing process. Constraints on agreement remain valid, for example for relative or adjectival attachment.

4 Parsing with the Grammar

One of the problems one encounters with parsing using a high level grammar is the multiplicity of (valid) analyses one gets as a result. While syntactically correct, some of these analyses should be removed for semantic reasons or in a particular context. One of the challenges is to reduce the parse number, without affecting the relevance of the results and without removing the desired parses. There are several ways to perform such a task, as described for example in (Segond and Copperman, 1997); we show here that finite state preprocessing for compounds is compatible with other possibilities.

4.1 Experiment and Results

The experiment reported here is very simple: it consists of parsing the technical corpus before and after integration of the morphological terms in the preprocessing components, using exactly the same grammar rules, and comparing the results obtained. As the compounds are mainly nominal, they will be analyzed just as regular nouns by the grammar rules. For example, if we parse the NP:

(7) *La boîte de vitesse (the gearbox)*

before integration we get the structures shown in Fig.3, and after integration we get the simple structures shown in Fig.4. The following tables show the results obtained on the whole corpus:

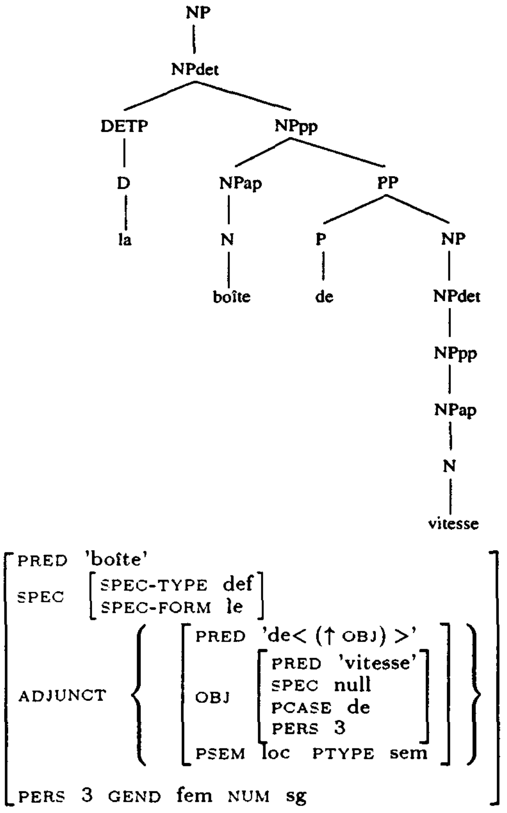

Figure 3: Before Terminology Integration

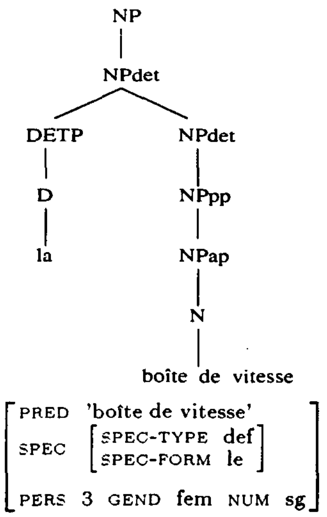

Figure 4: After Terminology Integration

- Before Terminology Integration:

Number of sentences	Token Average	Parse average	Time Average
with terms 358	10.59	4.21	1.706
without terms 384	8.98	3.77	1.025

- After Terminology Integration:

Number of sentences	Token average	Parse average	Time Average
with terms 358	8.86	2.79	0.987
without terms 384	8.98	3.77	1.025

The results are straightforward: one observes a significant reduction in the number of parses as well as in the parsing time, and no change at all for sentences which do not contain technical terms. Looking closer at the results shows that the parses ruled out by this method are semantically undesirable. We discuss these results in the next section.

4.2 Analysis of Results

The good results we obtained in terms of parse number and parsing time reduction were predictable. As the nominal terminology groups nouns, prepositional phrases and adjectival phrases together in lexical units, there is a significant reduction of the number of attachments. For example, the adjective *hydraulique* in the sentence:

(8) *Le voyant de levier de distributeur hydraulique s'allume. (The control valve lever warning light comes on.)*

can syntactically attach to *voyant*, *levier*, and *distributeur* which leads to 3 analyses. But in the domain the corpus is concerned with, *distributeur hydraulique* is a term. Parsing it as a nominal unit gives only one parse, which is the desired one. Moreover, grouping terms in unit resolves some lexical ambiguity in the preprocessing stage: for example, in *ceinture de sécurité*, the word *ceinture* is a noun but may be a verb in other contexts. Parsing *ceinture de sécurité* as a nominal term avoids further syntactic disambiguation.

Of course, one has to be very careful with the terminology integration in order to prevent a loss of valid analyses. In this experiment, no valid analyses were ruled out, because the semi-automatic method we used for extraction and integration allowed us to choose accurate terms. The reduction in the number of attachments is the main source of the decrease in the number of parses.

As the number of attachments and of lexical ambiguities decreases, the number of grammar rules applied to compute the results decreases as well. The parsing time is reduced as a consequence.

The gain of efficiency is interesting in this approach, but perhaps more valuable is the perspicuity of the results. For example. in a translation application it is clear that the representation given in Fig. 4, is more relevant and directly exploitable than the one given in Fig. 3, because in this case there is a direct mapping between the semantic predicate in French and English.

5 Conclusion and possible extensions

The experiment presented in this paper shows the advantage of treating terms as single tokens in the preprocessing stage of a parser. It is an example of interaction between low level finite-state tools and higher level grammars. Its shows the benefit from such a cooperation for the treatment of terminology and its implication on the syntactic parse results. One can imagine other interactions, for example, to use a “guesser”[3] transducer which can easily process unknown words, and give them plausible mophological analyses according to rules about productive endings.

There are ambiguity sources other than terminology, but this method of ambiguity reduction is compatible with others, and improves the perspicuity of the results. It has been shown to be valuable for other syntactic phenomena like time expressions, where local regular rules can compute the morphological variation of such expressions. In general, lexicalization of (fixed) multiword expressions, like complex preposition or adverbial phrases, compounds, dates, numerals, etc., is valuable for parsing because it avoids creation of “had hoc” and unproductive syntactic rules like *ADV → N Coord N* to parse *corps et âme (body and soul)*, and unusual lexicon entries like *fur* to get *au fur et à mesure (as one goes along)*. Ambiguity reduction and better relevance of results are direct consequences of such a treatment.

This experiment, which has been conducted on a small corpus containing few terms. will be extended with an automatic extraction and integration process on larger scale corpora and other languages.

[3] Already used in tagging applications

6 Acknowledgments

I would like to thanks my colleagues at XRCE, especially Max Copperman and Frédérique Segond for their help and valuable comments.

References

Salah Ait-Mokthar. 1997. Du texte ascii au texte lemmatisé : la présyntaxe en une seule étape. In *Proceedings TALN97*, Grenoble, France.

Joan Bresnan and Ronald M. Kaplan. 1982. *The Mental Representation of Grammatical Relations.* The MIT Press, Cambridge, MA.

Miriam Butt, Tracy Holloway King, Maria-Eugenia Niño, and Frédérique Segond. To appear. *A Grammar Writer's Cookbook.* CSLI Publications/ University of Chicago Press, Stanford University.

Jean-Pierre Chanod and Pasi Tapanainen. 1995. Tagging French - comparing a statistical and a constraint-based method. In *Proceedings of the Seventh Conference of the European Chapter*, pages 149–156, Dublin. Association for Computational Linguistic.

Jean-Pierre Chanod and Pasi Tapanainen. 1996. A non-deterministic tokeniser for finite-state parsing. In *Proceedings ECAI96*, Prague, Czech Republic.

Jean-Pierre Chanod. 1994. Finite-state composition of French verb morphology. Technical Report MLTT-004, Rank Xerox Research Centre, Grenoble.

Gregory Grefenstette and Pasi Tapanainen. 1994. What is a word, what is a sentence? problems of tokenisation. In *Proceedings of the Third International Conference on Computational Lexicography*, pages 79–87, Budapest. Research Institute for Linguistic Hungarian Academy of Sciences.

Christian Jacquemin. 1997. Variation terminologique : Reconnaissance et acquistion automatique de termes et de leur variante en corpus. Habilitation à diriger les recherches.

Lauri Kartunnen, Ronald M. Kaplan, and Annie Zaenen. 1992. Two-level morphology with composition. In *Proceedings of the 17h International Conference on Computational Linguistics (COLING '92)*, August.

Lauri Kartunnen. 1994. Constructing lexical transducers. In *Proceedings of the 19h International Conference on Computational Linguistics (COLING '94)*, August.

John T. Maxwell and Ron Kaplan. 1996. An efficient parser for LFG. In *Proceedings of LFG96*, Grenoble, France.

Julien Quint. 1997. Morphologie à deux niveaux des noms du français. Master thesis, Xerox European Research Centre, Grenoble.

Frédérique Segond and Max Copperman. 1997. Lexicon filtering. In *Proceedings of RANLP97*, Budapest.