Methods For Constructing Lexicon-Grammar Resources: The Example Of Measure Expressions

Matthieu Constant*

* University of Marne-la-Vallée
5,bld Descartes
Champs-sur-Marne
77454 Marne-la-Vallée Cedex 2
mconstant@univ-mlv.fr

Abstract

We construct, in the framework of the lexicon-grammar theory, a set of grammars dealing with measure expressions. First, we manually represent compounds with graphs: determiners such as *ten pounds of* and prepositions such as *34 miles from*. Then, by the means of lexicon-grammar matrices, graphs and a semi-automatic process, we build a set of grammars of kernel sentences e.g. *the door is 2-meter high*. Finally, we evaluate our methods and grammars according to three points: production, maintenance and concrete application.

1. Introduction

The representation of numerical expressions is a significant issue in Natural Language Processing (NLP) due to their very frequent occurrences in texts. In this paper, we build a set of grammars representing measure expressions in French and in English in the framework of lexicon-grammar methodology and with the aid of finite state technology. We first briefly present some points of the lexicon-grammar theory useful in our work (section 2). Then, we describe the process of constructing grammars of compounds in the form of graphs (section 3) and grammars of sentences (section 4). Finally, we evaluate our grammars (section 5).

2. Lexicon-Grammar Theory And Finite State Technology

2.1. Theoretical Bases

The Lexicon-Grammar is a systematic description of linguistic facts based on a transformational theory (Harris 1968; Gross 1975, 1994). The minimal unit of description is the elementary sentence. Predicates (verbs, nouns and adjectives) are systematically studied and encoded in Lexicon-Grammar Matrices (LGMs). Each lexical entry enters in an elementary surface structure according to which it is classified:

John eats a cake := N0 eat N1 [1]
Mary believes that John's wrong
:= N0 believe that S [2]
John makes a friend of Mary
:= N0 Vsup a friend of N1 [3]

Their transformational properties are systematically examined. Each lexical entry has its own behaviour, such as the French verbs *obéir* (to obey) and *penser* (to think) with the same elementary structure *N0 V à N1* (*V* for verb):

Luc obéit à Max = Luc lui obéit [4]
*Luc pense à Max = * Luc lui pense* [5]

Nominalization and adjectivization are also treated as in the following examples:

John analyses this problem
= John makes an analysis of this problem
Mary has courage
= Mary is courageous

In these cases, related predicates such as the noun *analysis* and the verb *to analyse* are encoded separately because they have different properties, but they are related to each other in the LGMs. Note also that constructions with noun predicates can be reduced into Noun Phrases (NPs): *Max approves the analysis of this problem by John.*

2.2. Linguistic Resources

Natural Language Processing (NLP) requires precise and large-scale linguistic databases. For this purpose, the informal European network of laboratories RELEX accumulates, in the framework of the Lexicon-grammar theory, linguistic components of three types in several languages (French, English, Portuguese, Italian, Korean, etc.): electronic dictionaries, local grammars and lexicon-grammar matrices (Leclère et al., 1991).

2.2.1. Electronic Dictionaries

Large-coverage morphological dictionaries of simple words have been built in order to recognize graphical words in electronic texts (Courtois & Silberztein, 1990). Inflection codes are associated to canonical entries to automatically generate their inflected forms. Compound words have also been encoded in compound dictionaries. These dictionaries are part of the DELA system. They are compressed in the form of Finite State Transducers (FSTs), therefore improving access performances.

2.2.2. Local Grammars

Local Grammars are in the form of graphs and describe local constraints (Gross, 1997). They are

[1] *Ni* stands fot the *i*th nominal agrument of a given predicate, where *i* is an integer.

[2] *S* stands for a sentence.

[3] Noun predicates enter in constructions with support verbs (*Vsup*).

[4] If S1 and S2 are sentences that have a transformational relation (in this case, pronominalization), they are equivalent in some ways and we write S1 = S2.

[5] * is the interdiction sign.

equivalent to FSTs after a few compiling operations: they can be seen as compound dictionary extensions. In terms of production and maintenance, this compact representation is a clear advantage in comparison with the list representation of electronic dictionaries. The use of sub-graphs makes it very modular. Local grammars can be applied to texts e.g. with the software INTEX (Silberztein, 1993) in order to recognize and tag utterances (cf. section 3).

2.2.3. Lexicon-Grammar Matrices

For each predicate (verbs, nouns and adjectives), syntactic properties and syntactic and semantic information on arguments are systematically encoded in the form of LGMs. Each column corresponds to a given property (e.g. pronominalization, passivation). Each row corresponds to one lexical entry. At the intersection of a row and a column, a plus sign indicates that the corresponding lexical entry has the corresponding property, a minus sign, that it has not; and finally, a string indicates lexical information (Cf. section 4.1). Part of the LGMs can automatically be transformed into FSTs e.g. to construct a syntactic parser (E. Roche, 1993, 1999). With each LGM entry, we associate a graph representing the set of its equivalent surface forms as shown in Figure 1[6]. The different paths of the graph are equivalent. Graph representation can be seen as a factorization of the frames and the slots in the EAGLES terminology (Barnett et al., 1996). The optionality of an argument is marked by the presence of an empty transition in parallel with the argument.

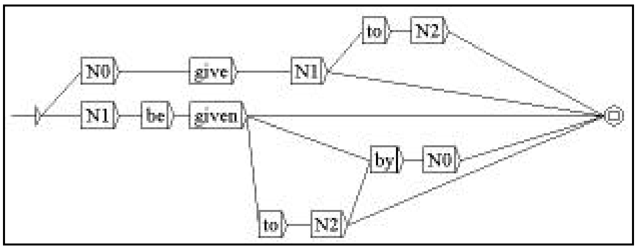

Figure 1: *give*

3. Compounds And Measure Expressions

3.1. Base Structure

Our first objective is to represent determiner phrases like *ten meters of* and compound locative prepositions like *ten miles from*, containing measure expressions of the form *Dnum Unit* (=: *fifty meters*). *Dnum* stands for a numerical determiner described in a set of graphs that recognize utterances such as *34.4* and *sixty-one* (Chrobot, 2000). *Unit* symbolizes units of measurement also represented in the form of graphs on the basis of Constant (2000) and describes utterances such as *meter*, *foot* plus their prefixed forms (e.g. *kilometer*), and their abbreviations (e.g. *ft).* Graph **NUMBER**[7] shown in Figure 2 is part of *Dnum* and represents sequences of digits (e.g. *123* and *1.7*). **GRAM**[8] (Figure 3) represents English weight units and has been established with the aid of an on-line dictionary of units of measurement (www.unc.edu/~rowlett/units/). It is included in *Unit*.

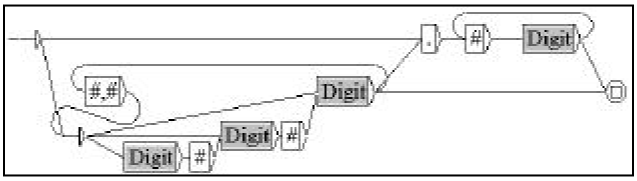

Figure 2: **NUMBER**

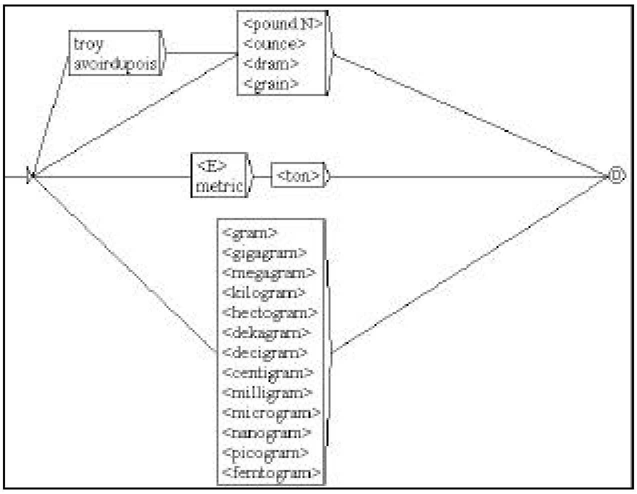

Figure 3: **GRAM**

We also need to use graphs of predeterminers (*PredDnum*) and postdeterminers (*PostDnum*) that occur frequently before and after *Dnum Unit*, such as in *environ 30 kilomètres* (about 30 kilometers) and *30 kilomètres environ*.

In English, currency units have a slightly different behaviour from other units: currency symbols are always located before *Dnum*: *£10*.

3.2. Determiner Phrases

We construct manually formal descriptions of compound determiners with the following basic forms:

(Det + E) Dnum Unit of =: *ten square meters of*
(Det + E) Dnum Unit de =: dix mètres carrés de

These determiner phrases have been studied in Buvet (1993). We propose a description with graphs. Their construction consists in assembling the graphs of section 3.1 and in choosing appropriate unit graphs (**GRAM** =: *gram*, **METER** =: *meter*, **METER2** =: *square meter*, **METER3** =: *cubic meter*, **DOLLAR** =: *dollar+ yen*, **DOLLAR_S** =: £, **SECOND** =: *second + year*). A simplified English graph is shown in Figure 4[9].

[6] This graph is a theoretical and simple example in order to help the reader's understanding.

[7] A symbol *s* in a grey box represents a call to graph *s*. Graph **Digit** recognizes numbers from 0 to 9. x#y means that the space character is forbidden between the words x and y.

[8] *<E>* is the empty symbol. For a given canonical form *x*, *<x>* stands for all inflected forms of *x* that are encoded in the dictionaries.

[9] Symbols in bold under boxes are the outputs of the graph.

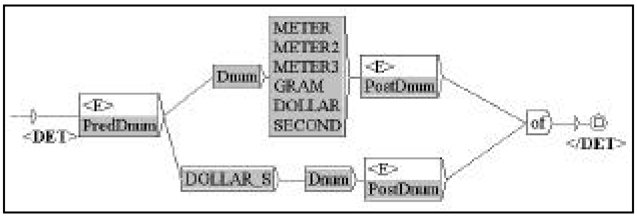

Figure 4: Measure determiner phrase

When the graph is applied to a text, the utterances recognized are automatically tagged as in the following example:

Independent States sold the European Union
<DET>3,000 tons of</DET> uranium

Semantic information could clearly be added in outputs because such compound determiners contain significant meaning. For example, *3,000 tons of uranium* can be related to the underlying sentence *the uranium has a weight of 3,000 tons* (Buvet, 1993, cf. section 4).

3.3. Locative Prepositions

We now describe semi-frozen locative prepositions (*Loc*) with the following basic forms:

Dnum METER from =: 30 kilometers from
à Dnum METRE de =: à 30 kilomètres de

They enter in elementary constructions with locative support verbs (Gross, 1996): *N0 Vsup Loc N1 (=: John is 30 kilometers from London).* This sentence can be semantically interpreted by the distance *d* between the geographical positions P0 and P1 of *N0* and *N1*: *d(John, London) = 30 km*[10]. Such an interpretation can be performed with FSTs containing variables (Silberztein, 1999) as shown in the theoretical graph in Figure 5. Let *u* be a sub-sequence of an utterance recognized by the graph. If *u* is recognized by a part of the graph between indexed parentheses (*i*), then *u* is indexed and is symbolized by the variable *$i*. Thus, if the utterance *John is ten meters from the swimming-pool* is recognized, the result of the interpretation is *d(John, the swimming-pool) = ten meters*.

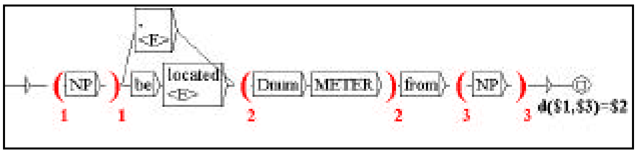

Figure 5: A semantic interpreter

The insertion of a direction involves the notion of vector such as in the sentences:

Mary is 10 km north of your house
The plane is 5km above John

In the case where P0 is within a circle the center of which is P1, *Loc* has the forms:

within a Dnum METER radius of
dans un rayon de Dnum METER autour de

We show in Figure 6 a simplified English graph of the locative prepositions containing the unit **METER**.

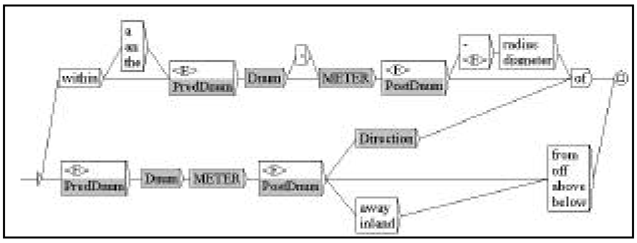

Figure 6: graph of locative prepositions

4. Sentences and measure expressions

4.1. Theory And Lexicon-Grammar Matrix Encoding

This section is based on a linguistic study by Giry-Schneider (1991). The elementary sentences we are concerned with have the following surface structures:

(1) *N0 have a N of Dnum Unit*
=: *Max has a weight of 80 kg*
(2) *N0 avoir un N de Dnum Unit*
= : *Max a un poids de 80 kg*

Each *N* has an appropriate set of units that can be represented by a graph. For example, graph **GRAM** describes the set of units appropriate to the noun *weight*. Constructions (3) and (4) are equivalent to (2):

(3) *N0 avoir Dnum Unit de N*
(4) *N0 avoir Dnum Unit de N-a*[11]
Cette corde a une longueur de 10 m
(This rope has a length of 10 m)
= *Cette corde a 10 m de (longueur + long)*

However, this permutation is not possible for all *N*s as shown in the following example:

Cette voiture a une vitesse de 10 km/h
(This car has a speed of 10 km per hour)
= * *Cette voiture a 10 km/h de vitesse*

The adjectivization transformation on elementary sentences (1) and (2) yields the following structures:

N0 is Dnum Unit N-a
=: *The rope is 10-meter long*
N0 être N-a de Dnum Unit
=: *la corde est longue de 10 m*

Sentences (1) and (2) can also be the result of a nominalization of the sentences:

(5) *N0 N-v Dnum Unit*[12]
= : *Le stylo coûte un euro*
=: *The pen costs one euro*

Constructions (1) and (2) can also be reduced into NPs of the form (6). In some cases, *N* can be zeroed as in (7).

(6) *N0 de un N de Dnum Unit*
= *N0 de Dnum Unit de (N + N-a)*

une porte d'une hauteur de 3 m

[10] *Unit* can be interpreted by the application of transducers that convert each unit into a standard one, e.g. *kilometre* *1000 m. Dnum* can be converted into a sequence of digits, such as *sixty-one* *61* (Chrobot 2000). Thus, *Dnum Unit* can be seen as a simple multiplication.

[11] *N-a* (=: *long*) is the adjective morphologically and semantically associated to *N* (=: *length*). Some *N*s have no *N-a*, e.g. *speed*.

[12] *N-v* (=: *to weigh*) is the verb morphologically and semantically associated to *N* (=: *weight*). Some *N*s have no *N-v*, e.g. *length*.

= une porte de 3 m de (hauteur + haut)
(7) *un homme d'une taille de 1,83 m*
= un homme de 1,83 m

From these observations, we build a LGM a selection of which is shown in Table 1. C1 indicates the *N*s described. C2 provides the names of the graphs representing the sets of units appropriate to *N*. C3 and C4 give respectively *N-a* and *N-v*. C5 indicates if *N* enters in construction (3). In our selection, the French word *tension* has two entries: the first one means *tension*, the other one means *blood pressure*. They have different appropriate units: the *blood pressure* has an empty unit symbolized by <*E*> while *volt* is associated to the *electric tension*. They also have different transformational properties:

Max a 13 de tension
** Cette ampoule a 60 V de tension*

Note also that we could add a new column in the LGM to indicate appropriate modifiers of *N*, such as *artérielle* (arterial) for *tension* (blood pressure) or *électrique* (electric) for the other *tension*.

C1	C2	C3	C4	C5
longueur	:METRE	long	-	+
Poids	:GRAMME	-	peser	-
Tension	:VOLT	-	-	-
Tension	<E>	-	-	+
Vitesse	:KMH	-	-	-

Table1: Measure sentence LGM

4.2. Reference Graph And Lexicon-Grammar Matrices

In section 2.2.3, it has been mentioned that an LGM can be semi-automatically transformed into graphs. A simple way of doing this is to build a reference graph that contains the set of all possible surface forms as shown in Figure 7. A transition @*i* (where *i* is an integer) is seen as a variable that refers to the *i*th column (or property) of the matrix. For each lexical entry (or each row), a new graph is automatically constructed from the reference graph by:

- removing the transition @*i* when the intersection of the *i*th column and the current row is '-'
- replacing @*i* by <*E*> (the empty element) when '+'
- replacing @*i* by the content of the intersection of the *i*th column and the current row, by default

The result of the process for the entry *longueur* is shown in Figure 8.

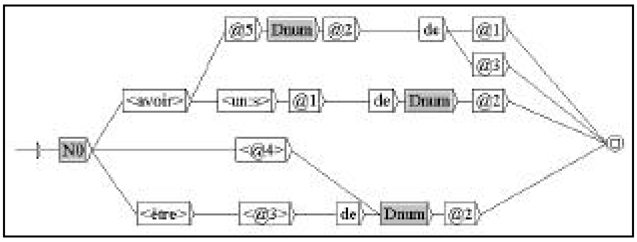

Figure 7: A sentence reference graph

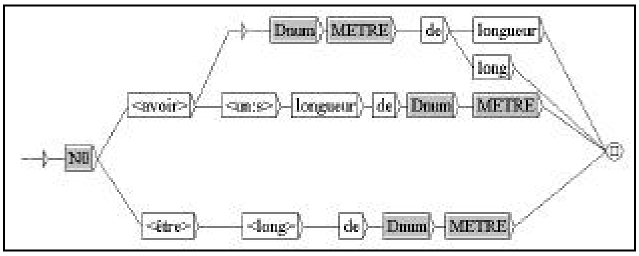

Figure 8: longueur

With the same process, we build reference graphs representing the *NP*s, reductions of the elementary sentences. Thus, with each *N*, we associate a *NP* graph.

5. Evaluation

We evaluate our methods and our grammars by taking into account three points: production, maintenance and application.

5.1. Production And Maintenance

Through the examples shown previously, it should be clear that graph formalism facilitates the production of grammars[13]. The application of our methods has to be manually controlled in order to construct precise lexicons based on linguistic facts: each lexical entry has a specific syntactic behaviour that cannot be entirely automatically predicted. Nevertheless, the use of automatic tools extracting information from large corpora makes this work less time-consuming. For example, in order to build our compound grammars, we built unit graphs with the aid of specific dictionaries, and then their application to a large corpus provided a list of occurrences. By examining their left and right contexts, we manually extracted relevant information e.g. the list of predeterminers. Furthermore, the use of LGMs, reference graphs and the semi-automatic process shown in section 4 avoids duplicating graphs by hand.

Maintenance is possible with our formalism. In graphs, the insertion of a new path is a very simple operation: inserting new transitions. The modification of a sentence grammar is also extremely simple and cheap. For example, the addition of a new property to a sentence grammar (e.g. appropriate modifiers) only requires adding a column in the LGM, modifying the reference graph and automatically generating the modified graphs for each lexical entry.

5.2. Concrete Application

The graphs of compounds shown previously are simplified and theoretical. Thus, we need to improve them in order to apply them to texts, which complicates graphs. We provide below a list of examples of improvement.

The theoretical sequence *Dnum Unit* used to describe the basic form of measure expressions does not exactly correspond to usage though most expressions of this type occur in texts. Each unit has its own syntax, e.g.

2 hours, 15 minutes and 2 seconds
5 feet and 2 inches

[13] Graphs are manipulated by the means of an editor (e.g. FSGraph in INTEX).

Thus, instead of dividing the sequence *Dnum Unit* into two graphs for each set of unit (e.g. Dnum METER, Dnum GRAM, etc.), we have to construct one graph *DnumUnit* (e.g. DnumMETER). As explained previously, these modifications in graphs are very simple to make. In column C2 of the Table 1, we replace *Unit* (=: GRAM) by *DnumUnit* (=: DnumGRAM) and then modify the reference graph.

The working corpus may also have a specific way of dealing with measure expressions. We shall adapt our grammars to it. For example, the electronic version of the newspaper *Herald Tribune* often juxtaposes measures in British units and their conversion into standard units, e.g. *It produced 1.2 million ounces (34,000 kilograms) of gold last year*.

Application to sentences is more difficult than to locally constrained expressions such as compounds because they rarely occur in their theoretical elementary form[14]. Measure expressions most of the time occur in the form of *NP*s and Adjective Phrases (*AP*s) transformationally related to the elementary sentence. With the aid of transformational properties, *NP*s and *AP*s, when recognized, are automatically related to their elementary sentences.

a 13-meter-long rope
une corde de 13 mètres de long
a five-day waiting period
une période d'attente de cinq jours
a 23-foot-long, three-ton hot *dog*

6. Conclusion

In this paper, we have described, in the framework of the lexicon-grammar theory, the construction of a set of finite state grammars of measure expressions in French and in English: compounds and sentences. We have also shown that our grammars based on the lexicon and the syntax is of great interest for the semantic interpretation of such expressions.

7. Acknowledgments

I would like to thank my director Eric Laporte for his help.

8. References

Barnett, R. et al. (1996). EAGLES recommendations on subcategorisation. EAGLES document EAG-CLWG-SYNLEX

Buvet, P.A. (1993). *Les déterminants nominaux quantifieurs*. Thèse de doctorat en linguistique. Paris, Université Paris 13.

Chrobot, A. (2000). Description des déterminants numéraux anglais par automates et transducteurs finis. In A. Dister (ed.), *Actes des Troisièmes Journées Intex* (pp. 101--118). Revue Informatique et Statistique dans les Sciences Humaines. Liège, Université de Liège.

Constant, M. (2000). Description d'expressions numériques en français. In A. Dister (ed.). *Actes des Troisièmes Journées Intex* (pp. 119--135). Revue Informatique et Statistique dans les Sciences Humaines. Liège, Université de Liège

Courtois, B. & Silberztein, M. (1990). *Les dictionnaires électroniques du français*. Langue Française 87. Paris: Larousse.

Giry-Schneider, J. (1991). Noms de grandeurs en avoir et noms d'unités. Cahiers de grammaire (pp. 29--49). Toulouse, Université de Toulouse-Le Mirail.

Gross, M. (1975). *Méthodes en syntaxe*. Paris: Hermann.

Gross, M. (1994). Constructing Lexicon-Grammars. In B.T.S. Atkins and A. Zampolli (eds.), *Computational Approaches to the Lexicon* (pp. 213--263). Oxford: Oxford University Press.

Gross, M. (1996). Les formes être Prép X du français. Lingvisticae Investigationes, XX:2, 217--270.

Gross, M. (1997). The Construction of Local Grammars, in E. Roche and Y. Schabes (Eds). *Finite State Language Processing* (pp. 329--352). Cambridge, Mass.: The MIT Press.

Harris, Z.S. (1968). *Mathematical Structures of Language*. New York: John Wiley and sons.

Leclère, C. & Subirats-Rüggeberg, C. (1991). A bibliography of studies on lexicon-grammar. LingvisticæInvestigationes, XV:2, 347--409.

Roche, E. (1993), *Analyse syntaxique transformationnelle du français par transducteurs et lexique-grammaire*. Thèse de doctorat en informatique, Université Paris 7.

Roche, E. (1999). Finite state transducers: parsing free and frozen sentences. In A. Kornai (ed.). *Extended finite state models of language* (pp. 108--121). Cambridge Press.

Silberztein, M. (1993). *Dictionnaires électroniques et analyse automatique de textes : Le système INTEX*. Paris: Masson.

Silberztein, M. (1999), Transducteurs pour le traitement automatique des textes. Rapport technique 57. LADL, Paris.

[14] In order to obtain better results for elementary sentence application, we shall, for example, insert new support verbs, adverbials between arguments, …