From Text to Knowledge*

M. Fernández[1], E. Villemonte de la Clergerie[2], M. Vilares[1]

[1] Department of Computer Science, University of Vigo
Campus As Lagoas s/n, 32004 Ourense, Spain
{mfgavilanes,vilares}@uvigo.es
[2] Institut National de Recherche en Informatique et en Automatique
Domaine de Voluceau, Rocquencourt, B.P. 105, 78153 Le Chesnay Cedex, France
Eric.De_La_Clergerie@inria.fr

Abstract. In this paper, we present a new approximation in Natural Language Processing (NLP) aimed at knowledge representation and acquisition using a formal syntactic frame. In practice, we introduce our implementation on an encyclopedic corpus in a botanic domain, illustrating the algorithm on a set of preliminary tests.

1 Introduction

Many documents written and published before the computer age are documents with an important content which is hard to search and utilize, because of the lack of automatic tools. To make the available information accessible in textual format, it is crucial to capture relevant data and convert them into a formal representation that will be used to help users. To do so, it is not feasible to recover the logical structure manually and, depending on the documents, it is a difficult task to consider automatic analyzers. This justifies the recent interest in automatic knowledge acquisition and, in particular, into applications on specific practical domains.

There are two main approaches in order to deal with the acquisition of semantic relations between terms. On the one hand, methods based on the comparison of syntactic contexts, often based on the use of statistic models. Briefly, some relations between terms are characterized by specific constructions that we can locate in texts from a particular lexical/syntactical pattern. Later, by using statistical calculus to compute the number of occurrences of the terms in these documents, we will be able to create a graph so as to extract specific collocations. The graph of co-occurrences are above all used to discover the sense and different applications of words through the detection of relevant structures in the graph, such as close noun sentences [1].

Other methods are grammar-oriented in the sense that they work directly on a parsing process, which serves as guideline for a more sophisticated linguistic

* Research partially supported by the Spanish Government under project TIN2004-07246-C03-01, and the Autonomous Government of Galicia under project PGIDIT05PXIC30501PN and the Network for Language Processing and Information Retrieval.

analysis. Texts are parsed in order to locate relations that can allow classes of words to be formed. In this way, the aim is to find similar terms that can later be grouped and classified in a graph. In this sense, some authors suggest conflating candidates that are variants of each other through a self-indexing procedure [4], while others [2] propose post-process parses so as to emphasize relationships between words.

We combine these two approaches in a proposal including original contributions. The acquisition phase is performed from a shallow parser, whose kernel is a *tree-adjoining grammar* (TAG) [5], a mildly context-sensitive formalism that improves parse recognition power in comparison to classic context-free grammars. The resulting partial parsing structures introduce ambiguities that we solve on the basis of an error-mining strategy [7]. Later, on the linguistic side, the French grammar used is compiled from a source *meta-grammar* (MG) [9], which is modular and hierarchically organized.

2 The running corpus

This research was conducted using a botanic corpus describing West African flora. We concentrate on the *"Flore du Cameroun"* published between 1963 and 2001, which includes about forty volumes in French, each volume running to about three hundred pages, organized as a sequence of sections, each one following a systematic structural schema and relating different species. So, sections are divided into a set of paragraphs enumerating morphological aspects such as color, size or form. This involves a nominal regime in sentences, named entities to denote dimensions, and also adverbs which modify the meaning of the verbs or adjectives so as to express frequency and intensity. The corpus[1] compiles typical vocabulary for the majority of the text based on this matter, and we consider it to be sufficiently representative for our purposes.

Our work forms part of the BIOTIM[2] [6] project on processing botanical corpora. We omit the initial phases of the project, related to the transfer from textual to electronic format [7] by means of an OCR, and also the capture of the logical structure of the text to browse it, through the combination of mark-up language and chunking tasks.

3 The parsing frame

Shallow parsing techniques for information extraction often rely on hand-crafted extracting patterns, a costly task that has to be redone for each new domain, that introduces ambiguities due to the high frequency of unknown terms.

In this context, the parsing frame we choose to work with is DyALog [8], taking TAG as grammatical formalism. We justify this on the basis that its dynamic programming architecture allows us to benefit from sharing of

[1] provided by the French Institute of Research for Cooperative Development.

[2] http://www-rocq.inria.fr/imedia/biotim/

parse computations and representations in dealing with non-determinism which improves efficiency. On the other hand, the consideration of TAG as grammatical formalism powers the system with new linguistic capabilities as, for example, cross and nested references as well as the constant growth property[3]. We do that by saving the valid prefix[4] and constant growth[5] properties and the polynomial complexity from context-free language. DyALog returns a total or partial parsing shared-forest on the basis of a TAG of large coverage for French. In this sense, the parser was improved, tailoring *meta-grammar* [6] and using error-mining techniques to track the words that occur more often than expected in unparsable sentences.

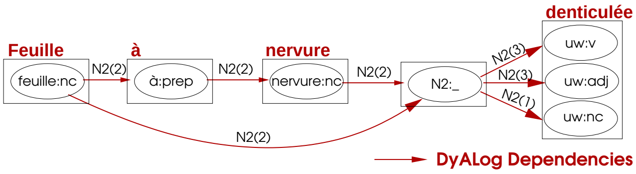

Fig. 1. Shared-parse dependencies from DyAlog

In Fig. 1, we illustrate the parse output for the sentence *"feuille à nervure denticulée"*, from now on our running example. The rectangular shapes represent *clusters*, that is, forms that refer to a position in the input string and all the possible lemmas with their corresponding lexical categories. We call these *nodes*, represented by ellipses. So, lexical ambiguities correspond to clusters containing nodes with different lemmas, or the same lemma associated to different lexical categories. Finally, arrows represent binary dependencies between words through some syntactic construction, showing syntactic ambiguities. So, the parse provides the mechanisms to deal with a posterior semantic phase of analysis, avoiding the elimination of syntactic data until we are sure it is unnecessary for knowledge acquisition.

3.1 Lexical ambiguities

The morpho-syntactic analysis of the sentences is performed by concatenating a number of different tasks compliant with the MAF proposal[7]. Its description is not the goal of this work, and we shall only focus on the operational purpose.

[3] It makes reference to linguistic intuition that sentences in a natural language can be built from a finite set of enclosed constructions by means of lineal operations.

[4] It guarantees that, as they read the input strings from left-to-right, the sub-strings read so far are valid prefixes for the language.

[5] It establishes the independence of each adjunction in a TAG with respect to others.

[6] which is modular and hierarchically organized

[7] http://atoll.inria.fr/catalogue.en.html#Lingua::MAF

In effect, in the context described, tagging becomes a non-deterministic and incomplete task, even if the vocabulary is relatively restricted such as in this encyclopedic corpus. Most of the common words used in these documents are nouns and adjectives which do not appear in the lexicon or dictionaries, causing them to be considered as unknown words. So, in our running example of Fig. 1, the word *"denticulée"* (`"dentate"`) has been assigned with a label unknown (`uw`), and several lexical categories are in competition following a grammar oriented approach. In fact, there are three possible associated lexical categories: verb (`v`), adjective (`adj`) and noun (`nc`). These ambiguities cannot always be solved at lexical level and should be left to be considered at parsing time, introducing an additional factor of syntactic ambiguity.

3.2 Syntactic ambiguities

Parsing in NLP with shallow/partial strategies often translates into only capturing local syntactic phenomena. This is an important drawback, because much of our information is expressed as nominal sentences, where the main noun is followed by prepositional attachments, as in the sentence *"feuille à nervure denticulée"*, in which we could locally consider two different interpretations: `"leaf with dentate vein"` or `"dentate leaf with vein"`. It becomes impossible here to establish if the word *"denticulée"*(`"dentate"`) relates to *"feuille"* (`"leaf"`) or to *"nervure"*(`"vein"`), as is shown in Fig. 1.

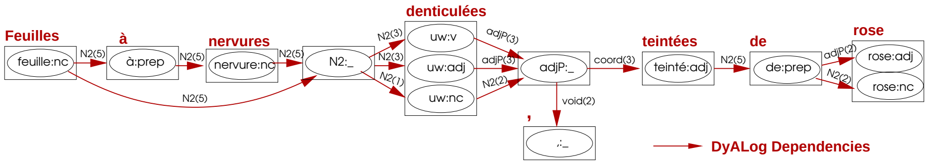

Fig. 2. Shared-parse dependencies from DyAlog

Another source of ambiguities is when the noun is followed by an enumeration of properties, i.e. coordination structures relating properties to a list of nouns, as in *"feuilles à nervures denticulées, teintées de rose"* (`"dentate leaf with vein, rose-tinted"`), the example shown in Fig. 2. Moreover, some of these properties may recursively be expressed as propositional sentences following the same structure. Therefore, we face non-determinism because of the attachment of a property to the main noun. For this reason, the above mentioned issues must be solved by relying on the use of a deep parser. Whichever the case is, the solution will lead us to use tools enabling us to go more deeply into the knowledge contained in the document.

4 Knowledge acquisition

Once we recover the shared-forest of dependencies for each total/partial parse generated by DyALog, we try to effectively extract the latent semantics in the document. In essence the idea is, by means of additional information from the corpus, to detect and eliminate useless dependencies. So, the lexical ambiguity shown in Fig. 1 should be decided in favor of the second translation `"dentate leaf with vein"`, because the word *”denticulée”* (`"dentate"`) is a typical characteristic that could only be applied on the word *”feuille”* (`"leaf"`) and not on the word *”nervure”* (`"vein"`). Furthermore, thanks to the encyclopedic corpus on botany, we could confirm this by examining it thoroughly. In this sense, to solve the ambiguity, we are looking for the information considering an iterative learning process to achieve our aim, in which term extraction constitutes the starting phase to formalize such a task.

1. $$P(\text{denticulée}, \text{adj})^{\text{cluster(j)}}_{\text{loc(0)}} = \frac{P(\text{denticulée}, \text{adj})^{\text{cluster(j)}}_{\text{loc}} * \#\text{arc}}{\Sigma_X P(\text{denticulée}, \text{X})^{\text{cluster(j)}}_{\text{loc}} * \#\text{arc}}$$

2. $$P(\text{denticulée}, \text{adj})_{\text{glob(n+1)}} = \frac{\Sigma^m_{j=1} P(\text{denticulée}, \text{adj})^{\text{cluster(j)}}_{\text{loc(n)}}}{\#\text{occ}}$$

3. $$P(\text{denticulée}, \text{adj})^{\text{cluster(j)}}_{\text{loc(n+1)}} = \frac{P(\text{denticulée}, \text{adj})^{\text{cluster(j)}}_{\text{loc(n)}} * P(\text{denticulée}, \text{adj})_{\text{glob(n+1)}}}{\Sigma_X \; P(\text{denticulée}, \text{X})^{\text{cluster(j)}}_{\text{loc(n)}} * P(\text{denticulée}, \text{X})_{\text{glob(n+1)}}}$$

Table 1. Learning the lexical category of *”denticulée”*

4.1 Term extraction

Before running the term extraction module, it is necessary to perform a previous step. The vocabulary used is quite limited. So, it is quite frequent to find words in the corpus which do not appear in the lexicon (unknown words). For those words many lexical categories could be suggested. Therefore, this step relies in adapting the error-mining principle to be able to identify the correct one, a protocol that can be also applied if we need to identify the most probable lexical category in polysemous known words. In these documents words tend to be monosemous, which enables us to have an idea of the most probable lexical category, but does not discard other alternatives.

Focusing on the word *”denticulée”* (`"dentate"`) in Fig. 1, we try to identify the correct interpretation by introducing an iterative process to compute the probability associated to each alternative in lexical category assignment. Following items in Table 1, we have the estimation of this probability associated to the adjective category. In Table 2 we can see different examples and the probabilities for all of these alternatives after applying 25 iterations. More in detail, we introduce this process as follows:

1. We compute the local probability of the lexical category of each word in sentences. To start the process, we take into account the weight of the initial probability of the lexical category considered, denoted by $\mathtt{P}_{\mathrm{loc}}$, that is a simple ratio on all possible lexical categories of a cluster in a sentence. We also stake into account the number of input derivations for each node with different lexical categories in a cluster, denoted by *#arc*. The normalization is given by the possible lexical categories involving the cluster in the sentence and represented by variable X.
2. Then, we re-introduce the local probabilities into the whole corpus in each cluster, in order to re-compute the weights of all lexical categories, estimating then globally the most probable ones. The normalization is given by the number of occurrences of this word with this lexical category, #occ.
3. The local value in the new iteration should take into account both the global preferences and the local injection of these preferences in the sentences, reinforcing the local probabilities. The normalization is given by previous local and global weights of the lexical category represented by variable X.

After a number of iterations, a fixed point assures the convergence of the strategy, as shown in Table 2, illustrating the global probabilities in the corpus obtained for the words *”denticulée”* ("dentate"), *”nervilles”* ("little veins"), *”obtusément”* ("obtusely") and *”viscidie”* ("higher part of the soldered androecium and gynoecium").

Form-lemma	**Possible lex. cat.**	**Probabilities**

denticulée - uw
("dentate")
	adj
	nc, v
		adj=1
		nc=0; v=0

nervilles - uw
("little veins")
	nc
	adv
	adj
	np, v
		nc=0.94
		adv=0.04
		adj=0.04
		np=0; v=0

obtusément - uw
("obtusely")
	adv
	nc, adj, v
		adv=1
		nc=0; adj=0; v=0

viscidie - uw
("higher part of the soldered
androecium and gynoecium")
	nc
	v
	np, adj, adv
		nc=0.92
		v=0.08
		np=0; adj=0; adv=0

Table 2. Global lexical categories obtained after 25 iterations

Once we have computed the weights of the lexical categories for each word in the corpus, the next step is to generate new dependencies. For each parsed sentence, we identify generic lexical and/or syntactic patterns, through the existing shared-forest of dependencies. We look now for pairs referring to governor/governed relationships between words, from those represented by arrows in Figs. 1 and 2. In particular, because of the nominal regime, we are more interested in dependencies between nouns and adjectives. This justifies filtering those dependencies, as shown in Fig. 3, following the dotted lines. So, the word *”nervures”* ("veins") is connected to *”denticulées”* ("dentate"), considering it as an adjective. Furthermore, we are also interested in extracting dependencies between nouns through, for example, prepositions such as *”feuilles à nervures”* ("leaves with veins"). In many cases the second part, that is the governed word, is a subclass of the first one, that is the governor word. The same occurs with dependencies between adjectives such as *"teintées de rose"* `("rose-tinted")`, *"rose"* being an adjective.

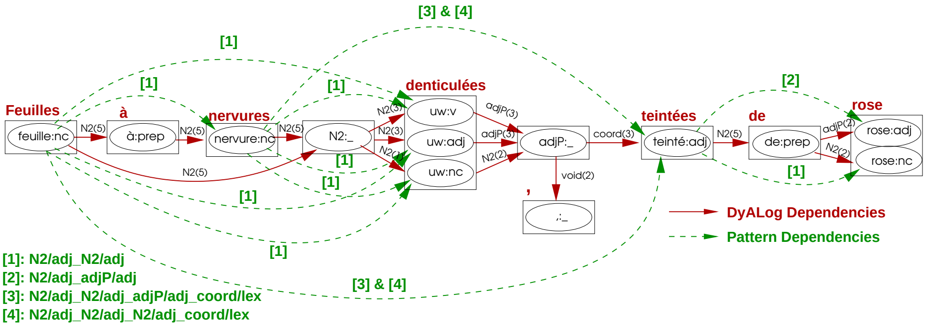

Fig. 3. New dependencies

The acquisition of semantic classes will be done on these new dependencies, since full cascading is not currently implemented because of its complexity and computational cost. The new dependencies extracted are either filtered or chains of dependencies that are completed by additional direct dependencies, for instance chains of coordination or enumeration. For example, in DyALog dependencies, the word *"denticulées"* `("dentate")` is connected with *"teintées"* `("tinted")` through an enumeration. Using our term extraction module, we can connect the word *"teintées"* `("tinted")` to other governor words such as *"feuilles"* `("leaves")` or *"nervures"* `("veins")`. Furthermore, we also filtered out some that we consider not very pertinent, such as adverbial modifiers. Now we are able to introduce our iterative knowledge acquisition process.

4.2 The iterative process

Although robust parsing makes it possible to detect syntactic patterns following the distributional hypothesis [3], we need to simplify the graph of dependencies resulting from the previous step in order to detect pertinent relations by trying to delete the spurious interpretations for the domain. In this sense, we establish a simple syntactic constraint: a governed word can only have one governor. It is sensible to think that a word can be related to only one, such as in Fig. 3, where *"teintées"* `("tinted")` could be governed by *"feuilles"* `("leaves")` or by *"nervures"* `("veins")` and, in consequence, we should eliminate one of these in the subsequent term clustering phase. This learning is based on the previous statistical approach.

The idea consists of combining two complementary iterative processes. On one hand, for a given iteration, we are looking for dependencies between governor/governed pairs that are considered most probable in the corpus. The probability of a dependency occurrence is labeled as `P(word1:c1,[label],word2:c2)`, being `word1` the governor word and `word2` the governed one, `c1` the lexical category of `word1` and `c2` that of `word2`, and `label` the tag of the dependency. In this sense, each probability of lexical categories is considered. On the other hand, the second process computes, from the former one, the most probable semantic class to be assigned to the terms involved.

Both are computed in a similar way as the example explained in Table 1. We need to know a local probability and a global one. In each iteration, we reintroduce these values in the local one and look for both semantic and syntactic disambiguation, each profiting from the other. At the end, a fixed point assures the convergence of the strategy [7].

5 Conclusion and Future work

The work described combines a number of NLP techniques in order to model concepts and relations between concepts contained in a text. Our aim is experimental and the goal is to introduce an architecture to generate a knowledge structure in order to develop question-answering facilities on textual documents.

In relation to previous proposals, we choose to work with a maximum degree of unsupervised tasks, which forces us to consider improved strategies in order to exactly identify both recurrent syntactic and stylistic patterns from text. The goal is to establish the linguistic context the parser will work with, in order to serve as a guideline for the later knowledge acquisition process.

References

1. O. Ferret. Using collocations for topic segmentation and link detection. In *Proc. of the 19th Int. Conf. on Computational Linguistics*, pages 1–7, vol 1, USA, 2002.
2. B. Habert, E. Naulleau, and A. Nazarenko. Symbolic word clustering for medium-size corpora. In *COLING*, pages 490–495, 1996.
3. Z.S. Harris. *Mathematical Structures of Languages.* J. Wiley & Sons, USA, 1968.
4. C. Jacquemin and D. Bourigault. Term extraction and automatic indexing. *Handbook of Computational Linguistics*, pages 599–615, 1999.
5. A.K. Joshi. An introduction to TAG. In *Mathematics of Language*, pages 87–114.
6. G. Rousse and É. Villemonte de La Clergerie. Analyse automatique de documents botaniques: le projet Biotim. In *Proc. of TIA'05*, pages 95–104, 2005.
7. B. Sagot and É. Villemonte de La Clergerie. Error mining in parsing results. In *Proc. of the 21st Int. Conf. on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 329–336, Australia, 2006.
8. É. Villemonte de La Clergerie. DyALog: a tabular logic programming based environment for NLP. In *Proc. of 2nd Int. Workshop on Constraint Solving and Language Processing (CSLP'05)*, Spain, 2005.
9. É. Villemonte de La Clergerie. From metagrammars to factorized TAG/TIG parsers. In *Proc. of IWPT'05*, pages 190–191, Canada, 2005.