TagParser: well on the way to ISO-TC37 conformance

Gil FRANCOPOULO
TAGMATICA
126 rue de Picpus
75012 PARIS FRANCE
gil.francopoulo@wanadoo.fr

Abstract

We present rapidly the family of standards that are currently under development within ISO-TC37. Then as an example of application of these ISO specifications for French, a concrete industrial parser is described: TagParser.

1 Introduction

The production, processing, use and re-use of linguistic data form a timely and costly part of the daily work of NLP industry and research teams.

Officially recognized specifications are needed. The ISO-TC37 work started from the GENELEX (Antoni-Lay), EAGLES, MILE (Bertagna) and TEI reports and we think that the family of specifications currently developed within ISO-TC37 is a good help as a common mechanism for fostering interchange of language resources and linguistic processing tools.

As a matter of fact, the title of this paper is not "TagParser: ISO-TC37 standard conformance" because most of the specifications developed within ISO-TC37 are not ISO standards yet. Let us recall that to be called "standard", a normative ISO document must have the status designated as "International Standard" with regards to the internal ISO process. An ISO document starts as a "Working Draft", then it becomes "Committee Draft", "Draft for International Standard", "Final Draft for International Standard" and finally "International Standard"[1]. At each step of the process, the document is balloted by the National Member Bodies (i.e. the organizations for normalization in each country), quite often technical comments are expressed and a new version of the document is produced taking into account the comments for the next round. The process is quite long and burdensome but the aim is to let the time in order to fully study the subject and reach a technical consensus.

In this paper, we describe a parsing scheme for French that has been developed during a period of twenty years (on a part time basis) and that has been recently modified according to the emerging TC37 specifications (Francopoulo 1988, 2005). A version for English has recently been developed and set up with the same strategy.

2 Interoperability requirements

Data associated with language resources are collected and stored in a wide variety of formats. These differences in approach inevitably lead to variations that prevent interchange and re-use of data.
Data are coded according to the following different levels:

- a physical level such as RDF schema (RDFS) or basic XML tags;

[1] see: www.iso.org + processes and procedures

- a basic constant level for values like character set representation (e.g. Unicode), country codes (e.g. ISO-3166), language coding (e.g. ISO-639-3), script coding (e.g. ISO-15924) that are already defined and stabilized outside ISO-TC37.
- a data category level for linguistic constants like /feminine/[2] and attribute values like /grammatical gender/;
- a structural level in which the organization of the data categories are determined in terms of classes and relations among classes. For instance, an entry in a lexicon will be coded according to a class specification holding an attribute called /part of speech/ and will be linked to one or several senses, that are themselves coded according to another class specification.
- a linguistic level such as annotation guidelines that specify what is the rationale that gives one or two NPs in the fragment: "Le député Robert Dupont …". Obviously, this is specific to a given language.
- the quality of the linguistic descriptions in terms of accuracy, coverage and depth with regards with the given concrete language.

Within ISO-TC37, a collective work is in progress in order to elaborate a family of specifications in order to improve current interoperability among language resources. But this work does not deal with all the levels mentioned hereby. The physical level is subject to debates and not fixed: so, one or several schemas are given in the informative annexes[3] of the different ISO standards. The basic constant level is already stabilized and widespread, thus these standards are just referred and, on purpose, no attempt is made to deviate or to redefine these values. Linguistic level and quality are considered as out of scope and thus, not addressed.

Data category and structure level are, on the contrary, the main focus of ISO-TC37 work by means of two kinds of normative objects:

- a data category registry (DCR) (Ide 2004)[4].
- a family of four structural specifications for lexicons and annotations.

TagParser has been re-engineered according to these standards, and seems to be one of the first parsing scheme for French that has this property.

One should note that to be interoperable, a parsing scheme does not need to directly implement all these standards. The TC37 specifications are designed as pivot formats implementing an abstract data model for lexicons and annotations. Thus, only mapping is needed. But why not using the ISO standards as direct guidelines? First, it is more simple to offer a direct interoperability instead of using mapping and secondly, the specification being rather generic, using them as a foundation offers a good guarantee for future evolution.

3 Data category level: data categories recorded in a registry

Data categories include both attribute such as /grammatical gender/ as well as a set of associated atomic values, such as /feminine/. In both cases, the abstraction behind an attribute or value is distinguished from its realization as some string of characters. To serve the needs of the widest possible user community, the DCR is developed with an eye toward multi-lingualism with the following criteria for each entry:

- an entry identifier;
- textual reference definitions which are expressed in various languages;
- names, possibly with synonyms, which are declared in various languages;
- possibly a shallow ontology is organized in order to link generic-specific values like /common noun/ vs. /noun/;
- possibly for some data categories dedicated to attributes, a range of permitted values.

An important property to mention is that data categories for lexicons and annotations are not separated. Of course, some values are specific to annotation, for instance, /punctuation/ that is mandatory

[2] Following ISO-12620 revision, data category identifiers are expressed between slashes.

[3] One should note that an ISO document comprises two sorts of sections: normative parts and informative parts, the latter being there only to help the understanding and usage of the normative parts.

[4] Data Category Registry: http://syntax.inist.fr

for annotation and is not for some lexicons. A second aspect deals with interoperability: with a set for lexicons and a set for annotation, the danger was too high to face a balkanization and thus to have incompatible sets.

Another point to mention is that the number of values is rather high, currently 600. Thus, the TC37/SC4 management decided to split the work into four sub-tasks on a linguistic basis and not on an object target basis.

So, four ISO profiles (each one corresponding to a sub-task) have been created:

- meta-data[5]
- morpho-syntax
- syntax
- semantics

And all these values are to be shared by lexicons and annotations. Currently (Fall 2007), a set of 600 data categories has been recorded in the ISO data category registry based on the work of:

- EAGLES for West-European languages;
- MULTEXT-East for East-European languages;
- Sfax University for Semitic languages;
- IMDI for meta data values;
- joint ISO-LIRICS-SIGSEM work and TimeML for semantic values;
- different TC37/SC4 works on lexicons and annotations for a small set of values.

One should note that two additional works are currently conducted (in Asia within the NEDO project for Asian languages and in South Africa for African languages) but the values are not yet entered in the database.

4 Structural level: a family of four specifications

So, the data categories provide a good foundation for interoperability between TC37 specifications and external formats, but, of course also among TC37 specifications.

The objects that we deal with are lexicons and annotations, the latest being either the result of a text hand-coding or the output produced automatically by a program. No distinction is made between the two types of annotations.

Four structural specifications are concerned:

- Lexical Markup Framework (LMF) that is the ISO specification for NLP lexicons (Francopoulo 2006)[6]. Individual instantiations of LMF may include monolingual, bilingual or multilingual resources. The same specifications are to be used for both small and large lexicons. The covered languages are not restricted to European languages but cover all natural languages. The descriptions range from morphology, syntax, and semantics to translation. An important part of LMF is dedicated to multilingual notations in order to both link senses of different languages, but also to control translation through a general ontology such as SUMO.
- Morpho-syntactic Annotation Framework (MAF) that first allows to segment a text into tokens and words, and secondly to mark these segments with values like /part of speech/ or /feminine/ (Clément 2005).
- Syntactic Annotation Framework (SynAF) that first rules how to delimit and mark syntactic phrases and sentences, and secondly rules how to describe relations between these phrases (Declerck 2006). SynAF annotations are built on top of MAF annotations. The sentence defines the boundaries of the fragments of textual documents to which SynAF applies.
- Semantic Annotation Framework (SemAF) that specifies how to add semantic marks to a text[7]. SemAF relies on MAF and SynAF.

[5] One should note that the term "meta-data" for this profile is a bit misleading because, in fact, all data categories may be used as meta-data. This profile covers management marks like /creation date/ and /author/.

[6] see: www.lexicalmarkupframework.org

[7] Contrary to MAF and SynAF, SemAF is a multipart specification and is not very well developed. Among the various parts that are scheduled, only the part one, that deals with time and events is active.

5 TagParser

5.1 Architecture

Two important points need to be mentioned in order to understand the parsing pipeline.
First, the parser itself is not a stand-alone program. The parser is just one of the components of a full parsing scheme that comprises modules like format guesser, format reader, language guesser, segmentation module, morphological analyzer, named entity recognizer, unknown word guesser and spelling corrector. All these modules implement a 'stand off' notation strategy as described in MAF ; **that is each module computes an annotation that is added layer by layer.** That means that the original textual content can be referred by means of pointers in all layers.
Secondly, TagParser proceeds in two main steps:

- a hybrid (symbolic and statistical) chunker that is corpus-based;
- a constraint solving module that is rule-based.

Oddly enough, there is no part of speech tagger. Using a tagger as a first pass for a parser is not very well suited for French. We know now that, since the GRACE campaign where 21 programs were compared with the objectives of tagging various sorts of texts. In this campaign, the winner was not a tagger but a robust chunker (Adda 1999, Vergne 2005). This can be explained by a certain number of reasons. One is that taggers usually operate on a window of two, three or four words, but in French, we have frequently various phenomena whose scope is broader. Another aspect is the significance of frozen multi-word expressions (MWE) that do not respect regular grammatical behavior, and so do not conform to a simple statistical model. The main problem for taggers in French is that they give too many wrong results. Ten years ago, when parsers had F-scores[8] in the range of 50 - 60%, this error rate was not a serious problem, but now, where parsers are more in the range of 70 - 90% or higher, this error rate is proportionally much more important. In the community, a familiar proverb is : "using a tagger for a parser is like starting to work by shooting oneself in the foot". This does not mean that statistical methods cannot be used for French. This just means that the notions of chunks and MWEs must be taken into account.

5.2 Lexicon

To this regard, an essential element is the lexicon. TagParser is associated with an LMF conforming lexicon comprising 600 K inflected forms obtained from 100 K simple lemmas and 30 K MWEs, these latest ones covering most frequent idiosyncrasies. The syntactic descriptions come mainly from DicoValence (Van den Eynde 2003). Here is an extract of the lexicon:

```
<LexicalResource dtdVersion="14">
   <GlobalInformation
      <feat att="languageCoding" val="ISO 639-3"/>
   </GlobalInformation>
   <Lexicon>
      <feat att="language" val="fra"/>
      <LexicalEntry paradigmPatterns="AsPassif">
        <feat att="partOfSpeech" val="adjective"/>
        <Lemma>
           <feat att="writtenForm" val="actif"/>
        </Lemma>
        <Sense id="S1">
           <feat att="definition"
                  val="Qui agit ou implique une activité"/>
           <SenseRelation targets="S3">
                 <feat att="label" val="antonym"/>
           </SenseRelation>
        </Sense>
        <Sense id="S2">
           <feat att="definition"
                 val="Propre à exprimer que le sujet est considéré comme agissant"/>
           <feat att="domain"    val="grammaire"/>
        </Sense>
        </LexicalEntry>
     <LexicalEntry paradigmPatterns="AsPassif">
        <feat att="partOfSpeech" val="adjective"/>
        <Lemma>
           <feat att="writtenForm" val="inactif"/>
        </Lemma>
        <Sense id="S3">
           <feat att="definition"
                  val="Qui n'a pas d'activité"/>
        </Sense>
        <Sense id="S4">
           <feat att="definition"
                 val="Qui n'a pas d'activité régulière, sans être chômeur"/>
           <feat att="domain"    val="juridique"/>
        </Sense>
        </LexicalEntry>
   …
</LexicalResource>
```

[8] The harmonic mean of precision (P) and recall (R): i.e. F-score = (2 * P * R) / (P + R)

The element "AsPassif" is a shared paradigm pattern defined elsewhere in the lexicon in order to describe that the lemma "actif" gives the inflected forms "actif", "actifs", "active" and "actives" for the four combinations of number and gender.

5.3 Coverage

Another aspect for a parser for a given language is to determine the corpus for this language. In France, we do not have any reference corpus like the British National Corpus or the American National Corpus for English. So, an attempt has been made to approximate a "balanced" corpus. The corpus is made of 82 M words: 65% of the texts comes from various news sources (belonging to general, sport and economic genres), 30% comes from parliamentary minutes (coming from both EC and French institutions), 4% comes from literary sources and 1% comes from emails and oral transcriptions of street dialogues. The proportions are rather open to criticism but this is all what we could collect. For us, this corpus is what we call "the French language". Let us add that this corpus is a raw corpus: there is no annotation of any kind.

5.4 Chunker development process

The main part of the parser is the chunker. This module has the difficult task of splitting the sentence into chunks, labeling these chunks and tagging part-of-speech ambiguities. The chunker produces only one solution.

The task of developing a rule-based chunker is a rather difficult one. The maintenance of a set of rules turns rapidly into a nightmare. We decided a long time ago to adopt a more stable strategy that is to induce a chunker from unordered examples. The question was how to select examples? The task of hand-coding annotation is a rather time consuming one. Thus, the annotation of randomly selected examples with the objectives of having a broad coverage is out of reach. The best strategy is dynamic annotation selection.

In this process, the parser is incrementally improved through a series of small steps.

Dynamic annotation selection algorithm: A tiny hand-coded corpus is used to serve as a bootstrap to build an initial parser by means of a machine learning algorithm. The parser is then applied to the raw corpus. Parsing failures are collected and the most simple failures are ranked. Similar situations are pruned. And the related sentences are then hand-coded and added to the hand-coded corpus. The system is then ready for another step (Francopoulo 2003).

In fact, it is a little bit more complex than that because the learning process has its own inner loop. Each time a new parser version is induced, an automatic check is made to ensure that the new version is at least better than the last one. This check is done by applying and evaluating the induced parser to the hand-coded corpus. Most of the time, the quality is better but if it is not the case, the situation is intellectually studied that may lead to lexicon accuracy improvement, tagset refinement or annotation guidelines modifications (that may lead themselves sometimes to backwards corpus updates). So, the process is globally incremental but some problematic situations may conduct to move temporarily one step behind, before going forward and further[9]. The algorithm pertains to the family of data-oriented parsing (DOP) when applied to chunks, on the contrary to DOP schemes applied to trees (Bod 2003).

The hand-coded corpus contains currently 90 K words and allows the parser to cover 96% of the raw corpus. Obviously, the hand-coded subset is not a corpus that is representative of the French language from a numerical point of view because the proportions are clearly biased. This corpus is more to be considered as a collection of difficulties.

The machine learning algorithm does not operate directly on data categories coming from the MAF result but on tagsets that are defined as combination of ISO data categories. Currently, the number of tagsets is 205. Most specific French grammatical words have their own tagset because these words exhibit rather specific combinations within phrases like NP or VP.

5.5 Constraint solving module

The machine learning mechanism is applied to chunks and works fine but, this strategy is difficult to apply to computation of syntactic relations because the annotated corpus is too small. A set of

[9] These notions are usually called local optimum vs. global optimum in the context of meta-heuristics like simulated annealing.

constraint rules have been hand-coded instead. The constraints combine grammars using strictly local rules with syntactic information obtained from DicoValence. The rules are organized into 14 packages, each of them implementing one of the relations as expressed in the PEAS guidelines for French[10]. A constraint solving module is applied during the parsing process.

5.6 Format of the result

Following MAF and SynAF, the result comprises six levels: Token, Word Form, Group, Relation, Sentence and Document.

A **token** is character string defined by an algorithm dedicated to segmentation.

A **word form** is defined as the result of :

- a named entity recognition,
- a look-up in the lexicon,
- or an unknown word guessing.

A **group** is a contiguous sequence of word forms. Aside from a limited number of specific situations[11], a group is the result of the chunking process. A group is non-recursive. The labels of the groups are taken from the DCR and the list is as follows:

- verbNucleus
- nounPhrase
- prepositionPhrase
- adjectivePhrase
- adverbPhrase
- prepositionVerbPhrase

A **relation** is a link between word forms and/or groups. The labels of the relations are also taken from the DCR and the list is as follows:

- subject
- auxiliary
- directObject
- verbComplement
- verbModifier
- complementizer
- attribute
- nounModifier
- adjectiveModifier
- adverbModifier
- prepositionModifier
- coordination
- apposition
- juxtaposition

A **sentence** is defined as the contiguous sequence of word forms linked by the transitive closure among relations.

A **document** is defined as the whole set of sentences in a file.

5.7 Implementation and speed

The code is written in Java for the development tools as well as for the parsing pipeline. Like most modern Java industrial codes, the multi-core and multi-processor features of recent computers are exploited when available. More precisely, the number of cores and processors is consulted at start-up time and accordingly, a certain number of parsing processes are run in parallel, the lexicon being loaded only once.

The learning phase together with the self-check phase takes 10 minutes. The whole parsing pipeline has a speed rate of 600 K words per hour on a server class machine (mono-Xeon quad-core). This speed is usually considered as acceptable for industrial purposes.

5.8 Evaluation

TagParser competes in the evaluation campaigns of the ANR-Passage project (see acknowledgements). The first evaluation will be conducted in December 2007 on a 'black box' basis.

The objectives of this project are also to build a 200 M words annotated corpus for French based on the combination of ten parser results. This project is a French National campaign that gathers most of the known parsers for French. The corpus will respect ISO-SynAF specifications, but it is still a bit too early to present any concrete result.

[10] see: www.limsi.fr/Recherche/CORVAL/easy

[11] In French, a chunk beginning with "de" cannot be distinguished as being NP ("Robert mange de la salade") compared to PP ("Robert arrive de la cuisine") from a syntactic computation based only on word constituency.

6 Conclusion

In this paper, we presented rapidly the family of standards that are currently under development within ISO by a great number of people coming from different countries.
Then, TagParser was described as an example of ISO specifications application.

Acknowledgements

This work was supported in part by the EU (eContent project 22236 LIRICS[12]) and in part by the French ANR-Passage project (Action ANR-06 MDCA-013[13]).

References

Adda G., Mariani J., Paroubek P., Rajman M. 1999 L'action GRACE d'évaluation de l'assignation des parties du discours pour le français. Langues vol-2.

Antoni-Lay M-H., Francopoulo G., Zaysser L. 1994 A generic model for reusable lexicons: the GENELEX project, Literary and Linguistic Computing 9(1): 47-54

Bod R. 2003 Extracting stochastic grammars from treebanks, in Treebanks: building and using parsed corpora, Abeillé ed, Kluwer

Bertagna F., Lenci A., Monachini M., Calzolari N. 2004 Content interoperability of lexical resources, open issues and MILE perspectives. LREC Lisbon

Clément L., de la Clergerie E. 2005 MAF: a morphosyntactic annotation framework. Language & Technology Conference Poznan

Declerck T. 2006 SynAF: towards a standard for syntactic annotation. LREC Genoa

Francopoulo G. 1988 A parser for French with induction of grammar rules, PhD dissertation, Paris-6 University

Francopoulo G. 2003 TagChunker: mécanisme de construction et évaluation. TALN Batz sur mer

Francopoulo G. 2005 TagParser et Technolangue-Easy TALN Dourdan

Francopoulo G., George M., Calzolari N., Monachini M., Bel N., Pet M., Soria C. 2006 Lexical Markup Framework (LMF) LREC Genoa

Ide N., Romary L. 2004 A registry of standard data categories for linguistic annotation. LREC Lisbon

Van Den Eynde K., Mertens P. 2003 La valence : l'approche pronominale, application au lexique verbal. Journal of French Language Studies 13, 63-104

Vergne J., Houden F. 2005 L'analyseur syntaxique Vergne-98 présenté aux actions d'évaluation GRACE et EASy. TALN Dourdan

[12] see: http://lirics.loria.fr
[13] see: http://atoll.inria.fr/passage