A Framework for Multilevel linguistic Annotations

Patrice Lopez
DFKI GmbH
Saarbrücken, Germany
lopez@dfki.de

Laurent Romary
LORIA
Vandœuvre-Lès-Nancy, France
romary@loria.fr

Abstract

This article presents a 3-step model for multilayer annotations of corpora. Each kind of annotation for a textual corporacorresponds to a different view on the same document. This principle can be expressed first with a general relational model dedicated to the organisation of LR. This abstract model is then implemented as an application of the XML formalism for the encoding of large corpora. The exploitation of this kind of annotated corpora requires efficient manipulation processes and reversive access. We propose to use a third step representation based on a set of optimised FSA resulting from the parsing of the XML documents. These propositions have been implemented in the first version of a workbench dedicated to the French *Le Monde* corpus.

1 Introduction

The majority of existing encoding solutions and tools are usually dedicated to one kind of annotation. In particular because there is a difficulty to incorporate different annotation schemes within one single hierarchy. Morever the growing size of available corpora makes them difficult to exploit and visualize. Besides, maintaining their structure is highly difficult when the annotations become complex.

The XML encoding formalism allows flexibility, portability and easy interchange of Linguistic Ressources (LR). Still XML has two main limits for the general encoding of complex multilevel LR:

- Complex structures cannot be easily represented as a single XML model and there is no general methodology to combine multiple XML hierarchies. What is needed is a way to represent an abstract view of the data to be encoded so that to fully capture the different relations among them.
- Although highly general and portable, XML does not comprise mechanisms to allow effecient access and retrieval mechanisms on the corresponding documents, such as those offered by a classical database. Moreover, to obtain a word given an annotated information can be very slow since the internal representations are not reversible.

The requirements for multilevel encoding of corpora are presented for example in (Bird and Liberman, 1999), (Cristea et al., 1998) and (Dybkjær et al., 1998). The different kinds of LR involved in the annotation schemes are generally stored in different XML documents and linked to a reference textual document (the corpus) resulting in an acyclic graph structure. This representation, in particular the relations it expresses, can be matched against the abstract model of a relational database to allow efficient store and access to the corresponding data. Indeed, the general workbench developped by the MATE project makes use of such a relational database for the internal representation of XML encoded information (Dybkjær et al., 1999).

The present paper intends to interleave XML and relational database approaches in order to obtain a general methodology and models for complex LR exchange and exploitation. We claim that (1) an additional abstract level similar to the one used in relational databases can be useful to define XML encoding principles and (2) a light relational database based on FSA inferred from the XML encoding can be particularly efficient for internal computation.

Applied to the multilevel annotation, the preliminary abstract model has to express the rela-

tions between the reference corpus and the different annotation levels. The solution we propose aims at associating the multilayer encoding of multiple views on a same corpus and the encoding of information redundancies. These redundoncies obtained from the XML structure would allow the design of internal representations, which can be optimized in time and space on the basis of FSA (Finite State Automata). We argue that a high level of structural organization of the LR is likely to lead to efficient processing, through the identification of similar factors and shared properties.

In the next section, we introduce our preliminary abstract level and present the model for multilevel annotations of textual corpora. We then show how to yield an XML encoding scheme from this model. In section 4, we suggest an internal representation inferred from the XML encoding and based on FSA. Finally a first implementation of these principles, experimented with a corpus of newspaper articles (Le Monde), is described.

2 A general relational model for linguistic ressources organisation

2.1 The RROM

Our first level of representation is called RROM (Relational Ressource Organisation Model). A RROM is composed of a set of Ressource Entities (RE) and a set of relations between these entities. A RE corresponds to an *independent* and *abstract* type of data that is used in a NLP system (for example word, lemma or category). Given a set of ressources, *Independant data* means that this data is not the result of a set of relations between other RE. A RE is represented with a general name and is associated to a data type definition. An *instanciation* of a RE is a realization of this RE according to the corresponding data type specifications. In following figures, an RE is graphicaly represented with a square box.

The relations between entities used in this model are characterized by two couples of integers on each edge. Depending on the direction of the relation, this couple gives the arity of the relation with the RE given by the edge, by analogy to the couples on the edges used in relational databases entity/relation models. Two RE can also be in relation. A RROM can be graphically represented with diagrams describing which REs are related to one another. In these diagrams, a Ressource Relation (RR) is represented with ellipsis. We distinguish two kinds of edges: $Unary$ edges (single line) which indicate a single link relation and $n-edge$ (double line) which means that a relation can link n instatiations of a RE at the same time.

2.2 First example: Morphological lexicon

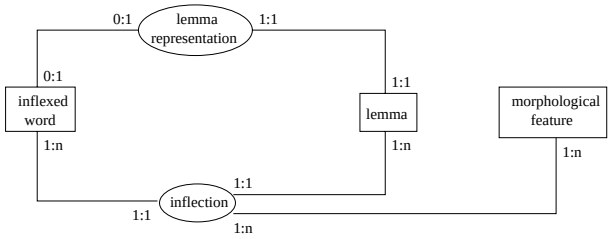

Figure 1: RROM for morphological lexicon.

A morphological lexicon database, as MULTEXT (Ide and Véronis, 1994), usually associates an inflected word to a set of lemmas and a set of features. Reversible access is needed for generation for example. A lemma is an abstract entity that is represented with a normal form of a word (the entry of a dictionary) and can be realized with all possible flexions of a word. We can distinguish as ressources entities inflected words, lemma and morphological features (including a category) that will characterize the inflection. An inflection is a relation between one inflected word, one lemma and a set of morphological features. Depending on the sense that one follows this inflection relation (from the lemma or from the inflected word), we obtain a reversible access. Each lemma is characterized by a link to one inflected word which is the normal form that identify this lemma (see figure 1). Respectively, an inflected word is not always the normal form of a lemma.

2.3 Second example: TAGML

TAGML (Tree Adjoining Grammars Markup Language) is a general norm for encodind and exchange ressources used with Lexicalised Tree Adjoining Grammars. A working group in France gathers people (mainly from TALaNa, ENST, INRIA Rocquencourt and LORIA) who work on this formalism and try to de-

fine standards for common grammars and grammar exchange, parsers, and tools developments. TAGML is an exemple of the high level of complexity of the ressources to encode. A LTAG grammar is defined by a morphological lexicon, a syntactic lexicon and a set of schemas (non lexicalized elementary tree paterns). The schema are ordered in tree families in order to capture generalities of lexicalizations given by the syntactic lexicon. Improvment of LTAG parsers and tools depends on how this huge amount of datas can be factorized in order to share computation[1].

The previous RROM model for morphological lexicon is extented to the other ressources needed at the syntactic level. An inflection (a lemma and a set of morphological features including verb mode for example) corresponds to a set of schemas. This lexicalization relation can include the instanciation of co-anchors (a lemma and a set of possibly underspecified morphological features) and of some additional syntactic features in the schema. Each syntactical instanciation give a complete elementary tree. If we assume that linguistic principles given in (Abeillé et al., 1990) and (Candito, 1999) are fullfilled by the grammar, each syntactical instanciation corresponds to only one semantic instanciation (semantic consistency principle). This model allows an incremental view of the lexicon ressources.

The figure 2 presents the corresponding RROM. To simplify, tree families and structuration of features are not included in this example.

2.4 Principles for multilevel annotated textual corpus

The principle of virtual ressources consists in describing a document as a set of elementary links (possibly unary) to subordinate documents. These links are occurences of a given information type which do not duplicate any content described in the documents which they are referring to. For instance a redundant *html* subdocument can be reached from different others *WWW* pages. The resulting representation is an acyclic graph which is relevant for the representation of multilayer annotations of textual

[1]See for example (Evans and Weir, 1998) for structure sharing. Similar sharing for features equations and derivation extraction are also possible.

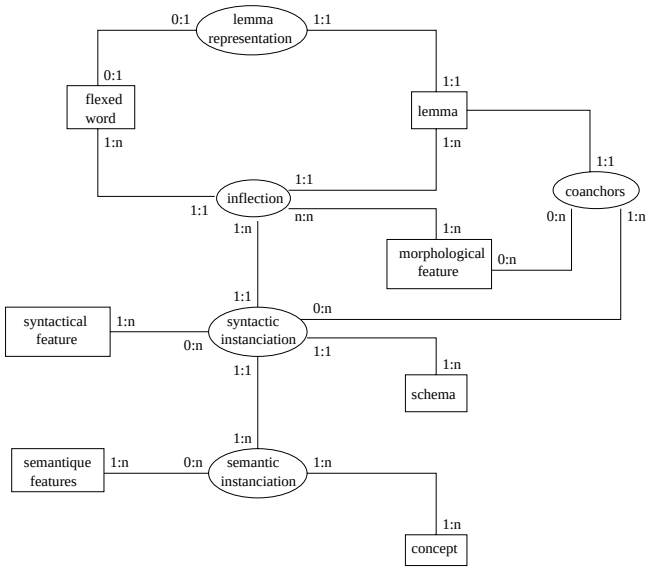

Figure 2: Simplified RROM for LTAG ressources.

corpora as shown (Bird and Liberman, 1999). Their abstract representation consists in a main axis refered to by different annotation levels by the way of edges.

In order to use this principle for multilevel linguistic annotation, we must identify correctly redundant subdocuments. We have realized this identification with a RROM. Each level of annotation (morpho-syntactic tags, phrase structure, refering expressions, dialogue acts, topics...) becomes a different view on the same text. Here, a particular annotation is a relation to a word or a sequence of reference words. We generalize this approach by considering the words as tags expressed in an independant subdocument. These word tags are then linked to a reference axis. Any kind of combined annotations, such as gestures or sounds, can be integrated according to this linking principle. This is particularly useful for the encoding of multimodal dialogues which may include gestures, visual scenes, speaking and reference representations.

The minimal unit of description of a textual corpus used to simulate the reference axis is the *event*. The events are ordered thanks to a strict order relation. An event corresponds to a point on a *reference axis* similar to the one of (Bird and Liberman, 1999) and can be identified with

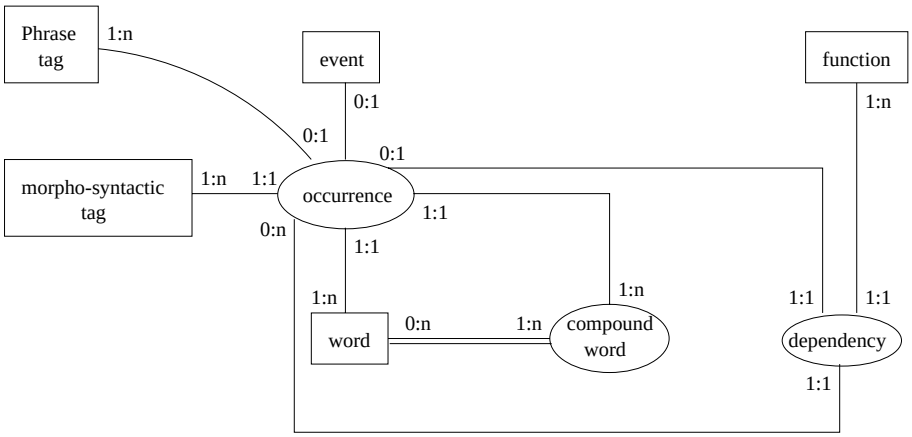

Figure 3: RROM for multilevel annotated textual corpus.

a *key*. This axis is similar to the one of (Bird and Liberman, 1999) and is adequate to represent a temporal axis. For each level of information, an annotation is a relation between a *tag* and one or more events. Each level of annotation imposes its own semantics on the link relations. A single link from an event to a word can be interpreted as an occurence of the word in a textual corpus. Two links to two events starting from a tag of gesture can signify the begining and the end of the gesture. Our experience has shown that a direct encoding of these links with the XML link machinery can result in documents which are difficult to develop, interpret and maintain. By using a RROM to represent the relations between various annotations and the reference axis, as shown in figure 3, we can express a multilevel annotation system with a precise comprehensive abstract model that will lead to an efficient use of virtual ressources.

In figure 3, we see that events are linked with occurrences of a word or a compound. Compounds can be also viewed as a relation between several words ($n - edge$). Each occurrence (of a single word or a compound) is linked with a morphosyntactic tag. Dependencies relation between two occurrences allow to obtain a full dependency tree. Finally a phrase tag can also be linked to an occurence relation in order to give the phrase category (VP, NP, ...) of the phrase dominated by this occurrence.

One can note the similarity between the RROM and the entity/relation models used in relational data bases. The main difference is that the links do not need contain their own attributes: The model can be realized with the current specifications of the XML norm. It can also use DTD for contraints expressions on ressources. Our proposition can be seen as an attempt to use the well specified methodology of relational databases with the portability, the expressivity and the adequacy of XML for textual datas and the efficiency of finite state representation for internal computation.

3 XML encoding for Multilevel annotated corpora

XML encoding and tools have several advantages compared with databases: Standardization aspects (data exchanges, unicode), existing dedicated tools (parsers, style sheet for document conversion), inheritance of the properties of SGML (textual ressources and linguistic oriented formalism, header specification, Text Encoding Initiative (TEI) specifications, ...). Moreover XML includes now interesting structuration features thanks to XML links and XML path specification.

The second step of our representation, the XML encoding, results directly from the organisation model of ressources. Classically, each annotation is composed by a main tag and a list of attributes. Each annotation is identified with a single identifier (*id*). The whole set of possible annotations for a given RE are gathered in a single document called *auxiliary ressource document*. For each RE we have one auxiliary ressource document that represents only

```
the:D
suitable:A
settings:Np
of:P
the:D
system:Ns
,:PONCT
since:ConjS
...
```

lexicon.xml

```
<w id="w0">the</w>
<w id="w1">suitable</w>
<w id="w2">settings</w>
<w id="w3">of</w>
<w id="w4">the</w>
<w id="w5">system</w>
...
```

tag.xml

```
<t id="t0">D</t>
<t id="t1">A</t>
<t id="t2">Np</t>
<t id="t3">P</t>
<t id="t4">Ns</t>
...
```

function.xml

```
<f id="f0">subject</t>
<f id="f1">modifier</t>
<f id="f2">determiner</t>
<f id="f3">noun noun complement</t>
<f id="f4">S</t>
...
```

phrase-tag.xml

```
<pt id="pt0">NP</t>
<pt id="pt1">VP</t>
<pt id="pt2">PP</t>
<pt id="pt3">AP</t>
<pt id="pt4">S</t>
...
```

Table 1: Example of a classical textual annotation and XML documents for Ressource Entities.

occurrence relations (occurrence.xml)

```
<link id="l0" event="0000" targets="lexicon.xml#w0 tag.xml#t0" />
<link id="l1" event="0001" targets="lexicon.xml#w1 tags.xml#t1" />
<link id="l2" event="0002" targets="lexicon.xml#w2
       tag.xml#t2 phrase-tag.xml#pt0" />
<link id="l3" event="0003" targets="lexicon.xml#w3 tag.xml#t3" />
<link id="l4" event="0004" targets="lexicon.xml#w4 tag.xml#t0
       phrase-tag.xml#2" />
...
```

dependency relations (dependency.xml)

```
<link id="0" targets="occurrence.xml#l0
       occurrence.xml#l2 function.xml#f2" />
<link id="1" targets="occurrence.xml#l1
       occurrence.xml#l2 function.xml#f1" />
<link id="2" targets="occurrence.xml#l3
       occurrence.xml#l2 function.xml#f3" />
...
```

Table 2: XML documents for Ressource Relations.

one time all the tags necessary for the corpus annotation. For each annotation level, we have a relational document which specifies the links between the tags given in the different auxiliary document and possibly the events of the reference axis. These documents are realized with the XML *link* tags which links the identifiers of the ressources tags with the keys (an integer here) of the event in relation. The reference axis is not represented explicitly by a document, but is given here implicitly by the list of event identifiers.

Each auxiliary ressource document supposes that a DTD (Document Type Definition) specifies the constraints on the expression of the corresponding annotation ressources. the XML encoding Given the relation model introduced previously and one DTD for each RE, we can specify in a unique way the corresponding XML encoding.

The table 1 gives an example of a classical annotated corpus and the new encoding documents. Each RE is encoded in an independant XML document: The document corresponding to the RE *word* of figure 3 can be view as the dictionnary of the corpus, the document *morphosyntactic tag* as the tag set. An additional XML document, not shown here, gives an explicit labelling for each tag (for instance *Np* stands for *plural noun*). Each element of these documents are identified in order to be linked. Each RR is also defined in a XML document using XML links as shown table 2.

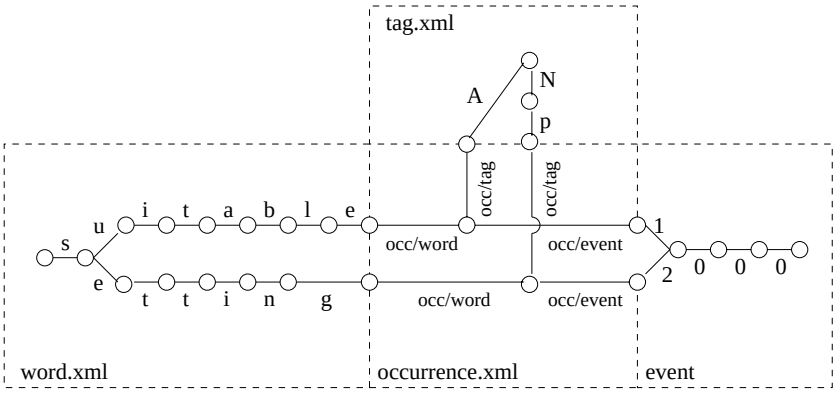

Figure 4: Internal Finite State Representation.

4 Efficient internal representations of the XML documents

The usual result of a XML parser needs a lot of memory and present generally inefficient access mecanisms for links. Event-based XML parser SAX allows to access to relevant datas without loading the full document, but is slower than a classical parser and has the same drawbacks concerning reversibility of access. Our proposal is to use FSA techniques for the internal representation of the XML document. FSA techniques present time and space optimisations and efficient reversible access. The efficiency of this representation exploit the redundancy of the information. By indentifying and encoding this redundancy with respectively a RROM and a XML document, we can obtain in a straightforward way this efficient internal representation.

Each auxiliary ressource document is compiled into an automaton with prefix sharing (lexicographic trees). Each XML relational document gives the transitions between the different automata obtained with the auxiliary ressource document. A relational document can be compiled into a transducer which links some auxiliary ressources entries identified by their XML *id*. Edges of this tranducer are labelled with the couple of names in relation (see figure 4) in order to allow fully reversible access. The reading of any tag gives in linear time the link to all auxiliary ressources in relation to this tag. Identifiers used in the XML encoding are only used to build this representation.

In figure 4 the reading of an event key gives the access to all level of annotation which are linked to the corresponding event. Given a word, the access to the following word is just the word associated to the next event key. The access to a given word (or a given tag) results in a list of event keys corresponding to all occurrences of the word (or the tag) on the reference axis (ie in the corpus) still in linear time.

Even in the case of very large corpora, the size of auxiliary ressources automata is limited. On the contrary, the internal representation of the reference axis and of the different transitions to words and tags can be very large when there are millions of events. In this case, cache techniques with temporary files may be necessary.

5 A workbench for visualization and exploitation

The first version of a workbench implementing these principles has been developped in the context of a project called CALIN. This graphical workbench is currently specialized for the French *Le Monde* corpus developped at TALaNa (University of Paris VII). Still this tool could be easily adapted to other classical tagged corpora or treebanks. The workbench is writen in Java and uses the Silfide XML Parser [2] which support XML link and XML path specifications. The workbench allows to visualise the reference corpus, to access to annotated information (morphology and syntax) simply by clicking on words (see figure 5). Three different modes allow to access to annotations linked to the word, to the compound or the full sentence containing the selected word. Syntactic annotation can be

[2] http://www.loria.fr/projets/XSilfide/EN/sxp/

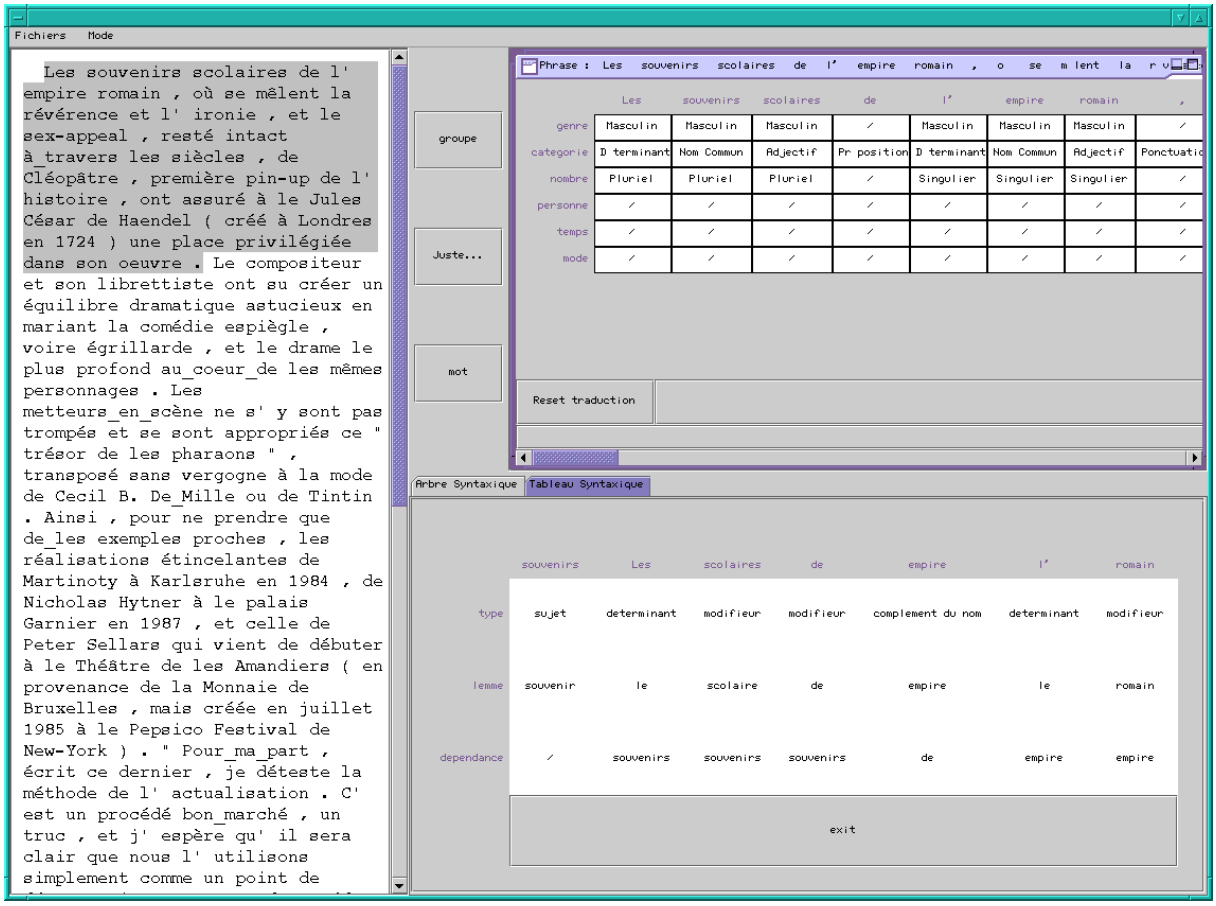

Figure 5: Screen shot of the workbench.

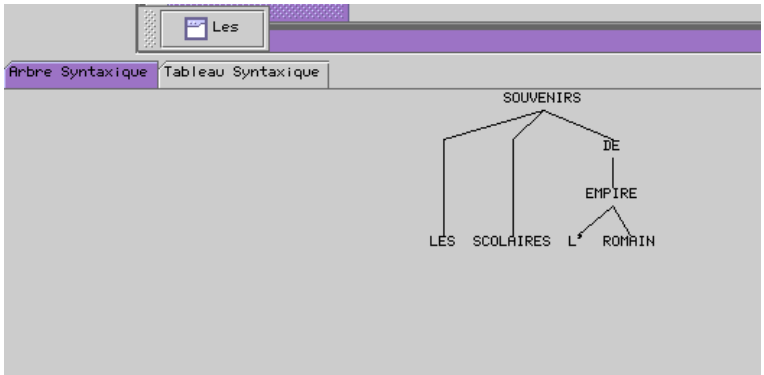

Figure 6: Dependency tree.

edited in a table or with a syntactic dependency tree (see figure 6)..

The implementation also provides a conversion tool to generate the XML documents from the existing annotated ASCII files as presented table 1. Currently the whole corpus is automatically divided into several XML documents which requieres less memory to be loaded by the XML parser than a complete XML document. Existing tools permits the conversion of the corpus in a single level of annotation in the proposed format. We can also project from our

XML encoding a particular level of annotation according to an existing standard XML annotation scheme.

6 Future work

Future works on the workbench during next steps of the CALIN project should include:

- Optimizations for the full *Le Monde* corpus (1 million of words)
- Search tools on words, tags or tree patterns using FSA based processing.
- Statistics and frequencies in the corpora.

Further research will try to formalize in our model the principle of *coding module* introduced in the MATE framework (Dybkjær et al., 1998). Our goal is then to integrate to the workbench functionalities such as:

- Merging several standard one-level annotations of a corpus into our multilevel framework.
- Projecting a particular level of annotation of a multi-level annotated corpus in an existing standard format.

Moreover we want to use the encoding principles proposed here to more complex multimodal annotated corpora resulting from a Wizard of Oz experience (combination of speech and gesture interaction). We also plan to study the uniform realization of the RROM model in XML on the basis of the *XML schema* proposals.

References

Anne Abeillé, Kathleen M. Bishop, Sharon Cote, and Yves Schabes. 1990. A Lexicalized Tree Adjoining Grammar for English. Technical Report MS-CIS-90-24, Departement of Computer and Information Science, University of Pennsylvania.

Steven Bird and Mark Liberman. 1999. A formal framework for linguistic annotation. Technical Report MS-CIS-99-01, Department of Computer and Information Science, University of Pennsylvania.

Marie-Hélène Candito. 1999. *Structuration d'une grammaire LTAG : application au français et à l'italien.* Ph.D. thesis, University of Paris 7.

Dan Cristea, Nancy Ide, and Laurent Romary. 1998. Marking-up multiple views on a text: Discourse and reference. In *Proceedings of ELREC 98.*

Laila Dybkjær, Niels Ole Bernsen, Hans Dybkjær, David McKelvie, and Andreas Mengel. 1998. The mate markup framework. Technical report, MATE delivrable D1.2, Odense University.

Laila Dybkjær, Morten Baun Moller, Niels Ole Bemsen, Jean Carletta, Amy Isard, Marion Klein, David McKelvie, and Andreas Mengel. 1999. The mate workbench. In *ACL'99 Software Demonstration Program, University of Maryland.*

Roger Evans and David Weir. 1998. A structure-sharing parser for lexicalized grammars. In *COLING-ALC*, Montréal, Canada.

Nancy Ide and Jean Véronis. 1994. Multext (multilingual tools and corpora). In *14th Conference on Computational Linguistics (COLING'94), Kyoto, Japan.*