EXCOM: an automatic annotation engine for semantic information

Djioua B.[1], Garcia-Flores J. [1], Blais A. [1], Desclés J-P. [1], Guibert G. [2], Jackiewicz A. [1], Le Priol F. [1], Nait-Baha L. [1], Sauzay B. [2]

[1] LaLICC, UMR 8139 Université Paris-Sorbonne/CNRS
28, rue Serpente, 75006 Paris, France
{bdjioua,jgflores,ablais,jpdescle,ajackiewicz}@paris4.sorbonne.fr
[2] France Télécom, Division Recherche & Développement
2 avenue Pierre Marzin, 22307 Lannion cedex, France
{gaelle.guibert,benoit.sauzay}@francetelecom.com

Abstract

In this position paper we describe the actual state of the development of an integrated set of tools (called EXCOM) for automatic semantic annotation. Annotation is generally used as an operation for marking textual segments to express some morphological and syntactic information. Establishing the semantic web on a large scale implies the widespread annotation of web documents with ontology-based knowledge markup. For this purpose, tools have been developed that allow for semi-automatic annotation of web documents with ontology-based metadata. This paper describes an automatic engine for semantic annotations based on linguistic knowledge and making use of XML technologies. We are persuaded that using linguistic information (especially the semantic organization of texts) can help retrieving information faster and better in the web. The basis aim of this engine is to construct automatically semantic metadata for texts that would allow us to search and extract data from texts annotated in that.

Introduction

In the context of the semantic web, electronic documents are marked up with metadata, using manual annotation with web-based knowledge representation languages such as RDF and DAML+OIL (Handschuh and Staab, 2003) for describing the content of a document. The aim of this work is to encourage the automatic annotation of electronic documents and to promote the development of annotation-aware applications such as content-based information presentation and retrieval.

Natural language applications, such as information extraction and machine translation, require a certain level of semantic analysis, which in practical terms means the annotation of each content segment with a semantic category (for a instance : definition, causality, citation or relation between named-entities).

EXCOM is an XML based system for an automatic annotation of texts according to semantic categories. The system allows us to express linguistic knowledge associated to semantic categories in a declarative way.

Our approach is based on the Contextual Exploration Method (Desclés et al, 1991, Desclés 1997), which states that semantic information associated to textual segments can be identified by linguistic primary marks (called indicators) and a set of clues that would help to tackle their polysemy. EXCOM associate a set of linguistic marks (indicator and complements lists) and declarative rules for each semantic annotation task. Conditions for executing a rule can be expressed in different ways, which switch on different levels of the engine. The result of a semantic annotation is a couple of documents, one for the structured text of the original text and other for the organized annotations for textual segments.

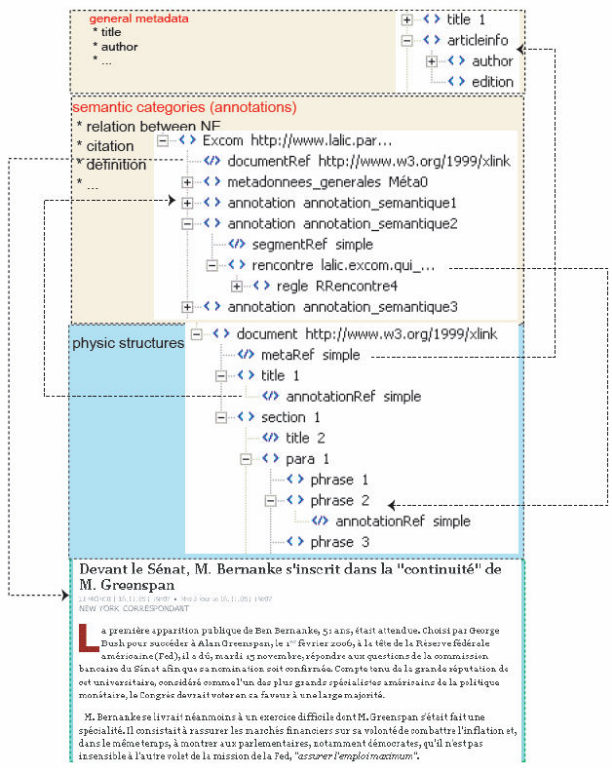
Linguistic annotation model

A major objective for EXCOM system is to explore the semantic of text in order to enhance information extraction and retrieval through automatic annotation of semantic relations. Most of linguistic-oriented annotation systems are based on morphological analysis, part-of-speech tagging, chunking, and dependency structure analysis. For example, the framework GATE (Cunningham and all, 2002) uses a set of reusable processing resources for common NLP tasks which are similar of those described above.

The methodology used by EXCOM, called Contextual Exploration, describes the discursive organisation of texts exclusively using linguistic knowledge present in the textual context. Linguistic knowledge is structured in form of lists of linguistic marks and declarative rules. The constitution of this linguistic knowledge is independent of a particular domain. Linguistic rules for identifying and semantically annotating segments use different strategies. Some of these rules use lists of simple patterns coded as regular expressions, others need to identify structures like titles, section, paragraphs and sentences for extraction purposes. The most relevant rules for EXCOM are those called Contextual Exploration (CE) rules. A CE rule is a complex algorithm based on a prime textual mark (called indicator), and secondary contextual clues intended to

confirm or invalidate the semantic value carried by the indicator.
The core of EXCOM annotation model is divides on several interlinked parts:

- Textual document
- General metadata like (title, author, edition …)
- Semantic annotations in relation with semantic categories



In this context, an annotation is considered as a set of XML/Xlink markup (Blanken H and al 2003, www.w3.org) related to a relationship defined between a textual (segment (a sentence for instance) and an instance of a semantic category (see page 5 for annotation examples).

The process of semantic annotation

The first step in constructing a linguistic categorization is to establish lists of semantic marks and contextual rules expressing a discursive notion (for instance, definition, citation and relationship between agents). The major subdivisions within a semantic categorization include:

- structural segments of the document (title, section, paragraph, sentence)
- linguistic marks (lexical, grammatical)
- search space (right and left context, an specific position in the document)
- indicator (verbs, prepositions, …)
- linguistic clues
- annotations (indicating the occurrence of a semantic category in a certain textual segment) ; for instance :
    - Connection relationships
        - Physical proximity
            - "Mr Hollande meets prime minister Blair".

The process of EXCOM annotation consists of the following steps (depicted in Fig.1):

Input: original text in HTML-XML-TXT format encoded on ISO-Latin1 or Unicode
Step 1[Pre-processing]: documents are converted to plain text format
Result: plain text format for document
Step 2[Segmentation]: plain text document transformed into structured document with structural annotations (title, section, paragraph and sentence)
Result: physical structures for document
Step 3[Annotation]: Process of annotation for a specific task :

Step 3.1[Regex rules]: Regular Expressions processing to identify first-level data (for instance, named-entities, locations, dates and temporal expressions).
Step 3.2[Structure rules]: Rules to identify complex structures based on first-leval annotation
Step 3.3[Contextual Exploration rules]: semantic rules processing with indicators and contextual clues for identifying a semantic category. This step is a complex process explained below.
Step 3.4[Negatives rules]: for identifying negations of semantic categories
Step 3.5[Modality rules]: to identify the achieved and possible semantic relations.

Result: Structured document and semantic annotation metadata.

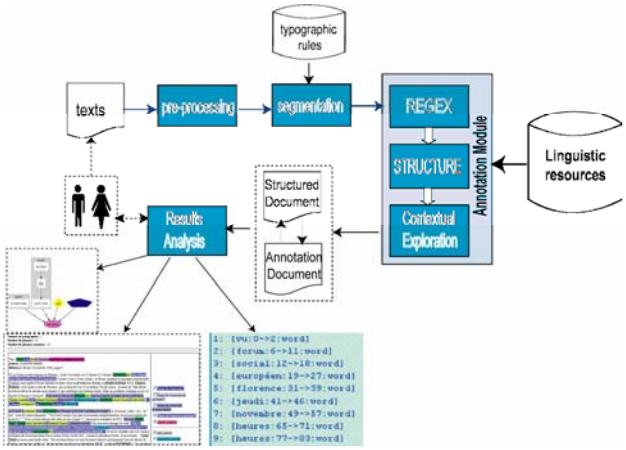


Fig 1: EXCOM architecture

The later one contains the result of the annotation process (annotated segments, annotation type, triggered rule,

indicator, linguistic clues) and the first one contains the source document structured by section, title, paragraph and sentence.

Post processing [Result analysis]: This last module for EXCOM allows us to explore the result of the annotation process, that is, the structured and annotated document.

EXCOM allows a user to explore an annotated document in three ways:

- A graphical interface to navigate into the document, with colourful annotations,
- A conceptual graph viewer, similar to dependency graphs.
- An indexation engine that makes possible the storage of annotated documents using a Nutch/Lucene API, for semantic information retrieval of web documents.

Linguistic resources

Linguistic resources are organized as typed semantic rules (contextual exploration rules, regular expression rules, etc.). Semantic rules are intended to capture the discursive organization of a text. Each rule is based on a set of markers lists, which can be used as indicators (to trigger the rule) or as clues (to confirm or refuse the annotation). List can be composed of lexical variations or regular expressions. These lists of linguistic terms are coded as Unicode plain files, while semantic rules are expressed in XML format.

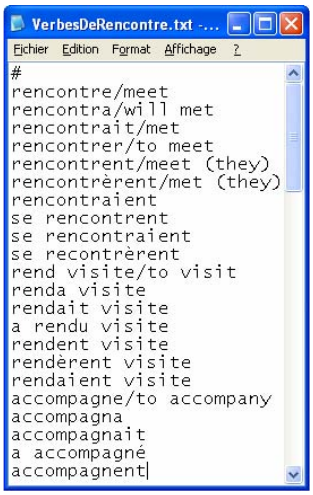


Fig 2: Markers lists

The annotation engine

The core of the annotation engine is organized on several layers interconnected (see Fig.1). The fist level layer (REGEX) encodes simple patterns as regular expressions. The second layer (STRUCTURE) allows the engine to trigger semantic rules from pre annotated segments. The third layer is charged of triggering Contextual Exploration rules. Let's see in detail these levels:

UNICODE: this layer allows the engine to perform multilingual processing for semantic annotation. All documents processed in platform EXCOM are coded in UTF-8.

REGEX: The first layer performs basic pattern annotation using a regular language. This level is used for named-entities identification, complex structures and some sub-segments that will be used as indicators or clues. Each regular expression can use basic, extended or advanced regular expressions capabilities (look-ahead, look-behind, Unicode patterns, etc ..) and can also call list of markers qualified by algebraic operators. EXCOM uses the regular expression engine of the Perl programming language. An example of annotation rule for named-entities.

```
<?xml version="1.0" encoding="UTF-8"?>
<regle id="EntitesNommees1" tâche="qui_ou_quand" point_de_vue="rencontre" type="regex"
  espace_de_recherche="phrase">
  <conditions>
    <motif type="regex" valeur="[A-Z][a-zàéèêâçôöúëñí]+"/>
    <motif type="regex" valeur="[A-Z][a-zàéèêâçôöúëñí]+"/>
  </conditions>
  <actions>
    <annotation type="ajout_element" annotation="nom_propre"/>
  </actions>
</regle>
```

This above rule represents an annotation of two contiguous words which starts with capital letter as a proper name (Jacques Chirac, Tony Blair, etc…).

STRUCTURE: this level makes possible to use pre-annotated segments as indicators or clues. This feature forces the engine to reach every annotated segment in the document structure (done with XPath expressions). For instance, if a user needs to annotation a textual segment like "the British Prime Minister Tony Blair" as a named entity using EXCOM, he should proceed within several steps:

(i) annotate nationalities (British) : markup <nationality>
(ii) annotate 'Prime Minister' as <office>
(iii) Annotate with a regular expression for proper name (markup <enamex>)
(iv) Combining these structures (markups) with the artcle 'the' to identify this complex named entity.

A semantic annotation rule for this segment is

```
<regle nom_regle="NamedEntity2" tache="qui_ou_quand" point_de_vue="rencontre" type="annotation
  simple" espace_de_recherche="phrase" ordre_motifs="suite">
  <conditions>
    <motif type="liste" valeur="art"/>
    <motif type="annotation" valeur="nationality"/>
    <motif type="annotation" valeur="office"/>
    <motif type="annotation" valeur="enamex"/>
  </conditions>
  <actions>
    <annotation type="ajout_attribut" espace="identique" nom_annotation="agent"/>
  </actions>
</regle>
```

CONTEXTUAL EXPORATION: This is the most important layer of the annotation engine. A CE triggers complex mechanisms that need the use of XSLT transformation language and a programming language (in this case, Perl). To continue with Prime Minister Blair, if a user wants to annotate a sentence like

"The British Prime Minister Tony Blair was in visit last week at Paris before …"

A semantic rule based on Contextual Exploration method would follow these steps:

(i) Express the semantic of the meeting category by means of a relevant indicator, represented in this sentence by the verb *'to be in visit'*

(ii) To confirm the indicator's "connection semantic", we need first to identify in the text the spatial expression *'at Paris'* in this right context

(iii) Indicator needs another expression like the named entity *'The british Prime Minister Tony Blair'* to allow the engine to annotate the sentence.

EXCOM uses an XSLT engine (with XPath parser) to identify nodes in the input XML document and process transformations by adding XML elements and attributes (see the below example).

Semantic categorization: the connection task

The semantic map presented here represents the various specifications of the semantic relation CONNECTION (who is with who).

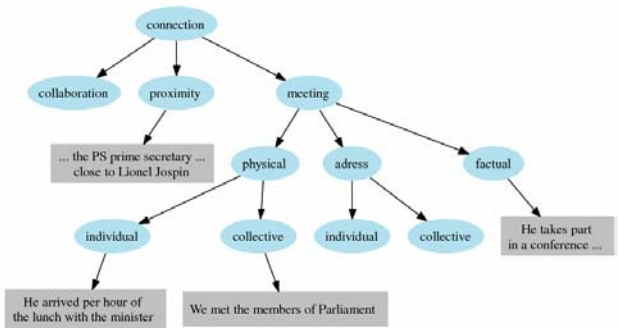


Fig 2: Semantic map for connection semantic category

The first level of the semantic map makes possible to release three types of meeting between agents: (i) *collaboration*, (ii) *proximity* and (iii) *general meeting*. Connection rules are triggered by occurrence of nouns connected to a meeting verb, and the semantic annotation is assigned if linguistic clues, like spatial prepositions, are found in the indicator's context. In addition, the process annotation must distinguish between a *generic annotation* and a *specific annotation*. In the *specific annotations*, linguistic rules use ENAMEX (proper nouns and named-entities or locations, like Prime Minister, Downing Street, etc.) and TIMEX (temporal expressions). In *generic annotation*, rules are declared according to Contextual Exploration for an indicator (generally a verb) on a textual segment within clues expressing a connection relationship.
Let's explain the annotation process through a linguistic declarative rule for identifying a meeting relation between named entities.

```
<!-- Visite de M. Hollande au Forum-->
<regle nom_regle="RRencontre101a" ordre_entre_indices="suite" tache="qui_ou_quand"
  point_de_vue="Rtitre" type="EC">
  <conditions>
    <indicateur espace_de_recherche="title" type="liste" valeur="NomsDeRencontre"/>
    <indice contexte="droit" espace_de_recherche="." type="liste" valeur="IndiceDetDeRencontre"/>
    <indice contexte="droit" espace_de_recherche="." type="annotation" valeur="nom_propre"/>
    <indice contexte="droit" espace_de_recherche="." type="annotation"
      valeur="expression_spatiale"/>
  </conditions>
  <actions>
    <annotation type="ajout_attribut" espace="identique" annotation="rencontre_evenementielle"/>
    <annotation type="ajout_attribut" espace="identique" degre_fiabilite="fort"/>
  </actions>
</regle>
```

The annotation engine process as follow:

(i) Identification of the indicator in the text (terms of the list *"NomsDeRencontre"* – a list of names like '*a visit'*) – this step generate an annotated and structured text with a markup `"<indicator rules='RRencontre101a'>Visite </indicator>"`. This process also generates an XML document which represents the candidates segments for this rule.

(ii) Generation of search spaces: parts of text where the engine will search the linguistic clues that will confirm or invalidate the indicator's connection value (one being a list form, the other two being the pre annotated segments named entities `<nom_propre>` and spatial expression `<expression_spatiale>`). These linguistic clues are identified sequentially (`ordre_entre_indices="suite"`). Only the right location is generated for this rule.

```
<title ID="1">
  <indicator rules="RRencontre101a"> Visite</indicator>
  <right_space>
    de
    <nom_propre>M. Hollande</nom_propre>
    <expression_temporelle> au Forum</expression_temporelle>
  </right_space>
</title>
```

(iii) Identification of the first term from the list *"IndiceDetRencontre"* – a name determinant – 'de/*of*' in this text.

```
<title ID="1">
   <indicator rules="RRencontre101a"> Visite</indicator>
   <right_space>
      <indice1 rule="RRencontre101a">de</indice1>
      <nom_propre>M. Hollande</nom_propre>
      <expression_temporelle> au Forum</expression_temporelle>
   </right_space>
</title>
```

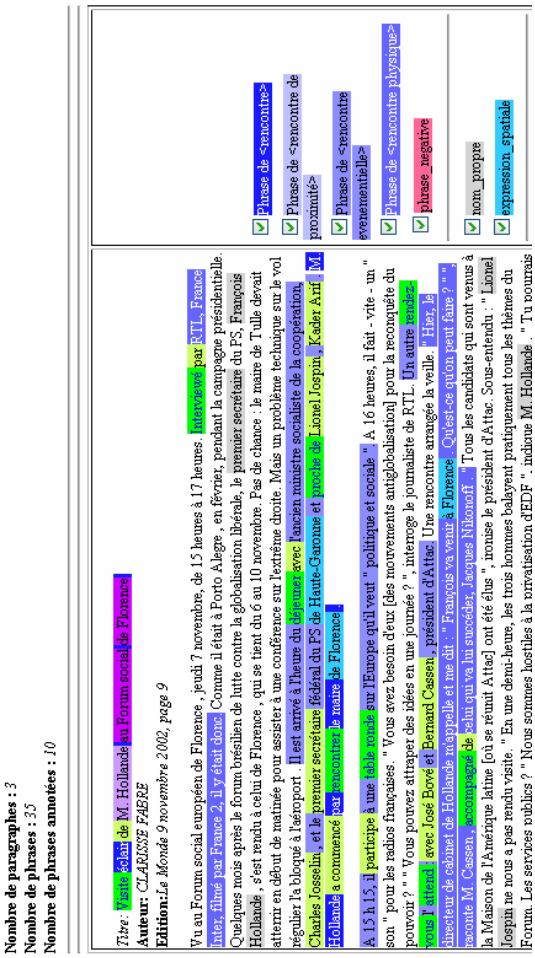
(iv) Identification of the second and third clues declared in the rule as a pre annotation of named entities (<nom_propre> and <expression_spatiale>). This operation is realized with an XML tree transformation using XPath/XSLT engine. A XSLT stylesheet is applied on the previous pre annotated XML document. This process produces two outputs: the structured document and its associated semantic metadata file.

(v) Annotation generation and relationship with the segment tile.

```
<title ID="1">
  <annotationRef
    xlink:type="simple"
    xlink:href="#annotation_sem1"/>
     Visite éclair de M. Hollande au Forum social
     de Florence
</title>
...
<annotation ID="annotation_sem1">
  <segmentRef
    xlink:type="simple"
    xlink:href="#1"
    type_segment="title"/>
  <rencontre
    libelle_annotation="lalic.excom.qui_ou_quand.
    Rtitre.rencontre_evenementielle"
    degre_fiabilite="fort">
    <nom_propre>M. Hollande</nom_propre>
    <expression_spatiale>
        au Forum social
    </expression_spatiale>
    <expression_spatiale>
        de Florence
    </expression_spatiale>
  </rencontre>
</annotation>
```

This annotation express that a phrase (the title of the news paper article) is marked as a connection relationship between named-entities, whose one of the agents is identified as “M. Hollande”.
EXCOM results are prepared with these two structures to be easily manipulated by final users towards graphic viewers. Programs can also use these two interconnected (XLinked structures) documents for information extraction and in an indexing processing.

The annotation process results are viewed for this example within HTML/CSS file like :



Related Work

EXCOM draws from a large pool of previous work on Contextual Exploration architectures and development environments for representing and processing scientist texts (Ben Hazez 2004, Crispino G. 2003). We have also drawn from work on representing texts with XML structures (DocBook and TEI grammars). The hierarchic layers of the annotation engine are drawn from the GATE framework architecture (Cunningham, 2002) – with an important distinction: using a new Contextual Exploration layer for semantic annotations in discourse analysis.

Conclusion and future work

EXCOM is a comprehensive framework for creating automatic semantic annotations based both on Contextual Exploration method and XML technologies. This current version is being tested with different semantic categorisations like "relations between named entities" as shown in this paper, localization relations, text summarization and control sentences extraction. EXCOM is the last implementation of "Contextual Exploration" system developed in the last ten years. This system benefits of the ContextO (Crispino, 2003) and Semantex (Ben Hazez, 2002) large experience. EXCOM's future developments will include multilingual automatic annotation for semantic categorizations (mainly in Korean and Arabic languages). Another important feature under development at the current stage is the link between semantic annotation and web documents indexation. This feature will allow us to perform real semantic oriented information retrieval, which could be the base for a different type of web search engine. Automatic semantic annotation generated by EXCOM could be used for making a semantic inverted index able to find relevant documents for queries like *"Who meets Tony Blair last year?"*

References

Ben Hazez 2004, Modèle de filtrage et de structuration des textes, Conception et réalisation d'une plate-forme orientée objet, Ph. D Thesis, Université Paris-Sorbonne,

Blanken H, Grabs T, Schek H-J, Schenkel R., Weikum G.,(Eds) 2003 Intelligent Search on XML Data, Applications, Languages, Models, Implementations, and Benchmarks, *Springer Verlag*,

Crispino G. 2003, Une plate-forme informatique de l'Exploration Contextuelle : modélisation, architecture et réalisation (ContextO). Application au filtrage sémantique de textes, Ph. D., Univ. Paris-Sorbonne

H. Cunningham, D. Maynard, K. Bontcheva, V. Tablan. GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02). Philadelphia, July 2002

Desclés Jean-Pierre 1997, Système d'exploration contextuelle, in Co-texte et calcul du sens, 215-232. Calif : eds Guimier, Presses Univ. Caen

Desclés J-P., Jouis C., Oh H-G., Reppert D. 1991, Exploration contextuelle et sémantique : un système expert qui trouve les valeurs sémantiques des temps de l'indicatif dans un texte., in Knowledge modeling and expertise transfert, Eds D. Hérin-Aime, R. Dieng, J-P. Regourd, J-P. Angoujard, 371-400. Calif : IOS Press,

Handschuh S., Staab S., 2003, Annotation for the Semantic Web, Volume 96 Frontiers in Artificial Intelligence and Applications, IOS Press

Mourad G. 2001. *Analyse informatique es signes typographiques pour la segmentation de textes et l'extraction automatique des citations. Réalisation des Applications informatiques : SegATex et CitaRE*, Ph. D Thesis, Univ, Paris-Sorbonne

XML, XPath, Xlink and XSLT recommendations: http://xmlfr.org/w3c/