Mining Biomedical Texts to Generate Semantic Annotations

Khaled Mohamed Khelif, Rose Dieng-Kuntz, Pascal Barbry

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Mining Biomedical Texts to Generate Semantic Annotations

Khaled Khelif – Rose Dieng-Kuntz – Pascal Barbry

N° 6102

January 2007

Thème Sym

ISSN 0249-6399

Mining biomedical texts to generate semantic annotations

Khaled Khelif[1], Rose Dieng-Kuntz[1], Pascal Barbry[2]

Thème Sym – Systèmes symboliques
Projet Acacia

Rapport de recherche n° 6102 – Janvier 2007 - 25 pages

Abstract: This report focuses on text mining in the biomedical domain for the generation of semantic annotations based on a formal model which is ontology. We start by exposing the generic methodology for the generation of annotations from texts. Then, we present a state of the art on different knowledge extraction techniques used on biomedical texts. We propose our approach based on Semantic Web Technologies and Natural Language Processing (NLP): it relies on formal ontologies to generate semantic annotations on scientific articles and on other knowledge sources (databases, experiment sheets). This approach can be extended to other domains requiring experiments and massive data analyses. Finally, we conclude with a discussion about our work and we present some learnt lessons.

Keywords: NLP, semantic annotations, knowledge acquisition, ontologies, life science, semantic web

[1] INRIA Sophia Antipolis - Acacia – 2004 route des lucioles, BP 93, FR-06902, Sophia Antipolis – Khaled.Khelif@inria.fr; Rose.Dieng@inria.fr
[2] IPMC - Institut de Pharmacologie Moléculaire et Cellulaire - 660 Route des Lucioles Sophia Antipolis 06560 VALBONNE ; barbry@ipmc.cnrs.fr

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles, BP 93, 06902 Sophia Antipolis Cedex (France)
Téléphone : +33 4 92 38 77 77 — Télécopie : +33 4 92 38 77 65

Fouiller les textes biomédicaux pour générer des annotations sémantiques

Résumé: Ce rapport s'intéresse à la fouille des textes dans le domaine biomédical afin de générer des annotations dites sémantiques du fait qu'elles sont basées sur un modèle formel qui est l'ontologie. Nous commençons par exposer la méthodologie générique pour la génération d'annotations à partir des textes. Ensuite, nous présentons un état de l'art sur les différentes techniques d'extraction de connaissances à partir des textes biomédicaux. Nous proposons notre approche, basée sur les technologies du web sémantique et du traitement automatique de la langue naturelle (TALN), et qui repose sur l'utilisation des ontologies pour la génération d'annotations sémantiques sur des articles scientifiques et d'autres sources de connaissances du domaine biomédical (base de données, cahiers d'expériences, etc.). Cette approche peut être généralisée à d'autres domaines requérant des expérimentations et traitant un grand flux de données. Enfin, nous concluons en discutant notre travail et en présentant quelques leçons appris.
Mots clés: TALN, annotations sémantiques, extraction de connaissances, ontologies, science de la vie, web sémantique

1 Introduction

In 2006 the MEDLINE database contains over 14 million citations, and this number is growing at the rate of 500,000 new citations each year. To help users to navigate and retrieve information in this vast and growing collection of documents, several methods and systems have been proposed to annotate these documents automatically by extracting information from texts.
Text mining offers to these systems and methods techniques to automatically extract relevant information contained in free text. Furthermore, in the semantic web context, the annotations generated are formalized by ontologies to ensure semantic interoperability between the extracted knowledge embedded in annotations and other knowledge sources.
In this document, we present briefly a generic methodology to generate ontology-based annotations using NLP techniques. Then, we will propose a synthesis on approaches proposed for text mining in biomedical domain.

2 Semantic annotation generation: the generic methodology

An annotation, or metadata, indicates “data about data”. In terms of documentation, it is secondary information affixed to a primary resource which is the document. In addition to simple information such as the title and the authors, a "semantic" annotation provides a more precise description of the knowledge contained in the document and its semantics in the domain. A semantic annotation must be well defined, easy to understand by the domain experts and not ambiguous. To fulfil these requirements, a semantic annotation should be based on a formal model of the domain (i.e an ontology).
The formalisation of the annotation scheme using the ontological hierarchy (i) enables annotators to choose the appropriate level of annotation detail, (ii) helps to constrain structure, to diminish ambiguity, and (iii) reduces errors in the annotation process. In addition, the fact that annotation is based on an ontology leads to use standard formalisms such as RDF [20] or OWL [21] which allow the reuse of these annotations by different annotation tools and search engines.
In spite of its advantages, the creation of semantic annotations is a difficult and time-consuming process for biologists. However, recent advances in natural language processing (NLP) techniques open new ways for automating the information extraction and the annotation generation task.
The figure 1 shows the data flow of the generation of a semantic annotation using the NLP techniques.

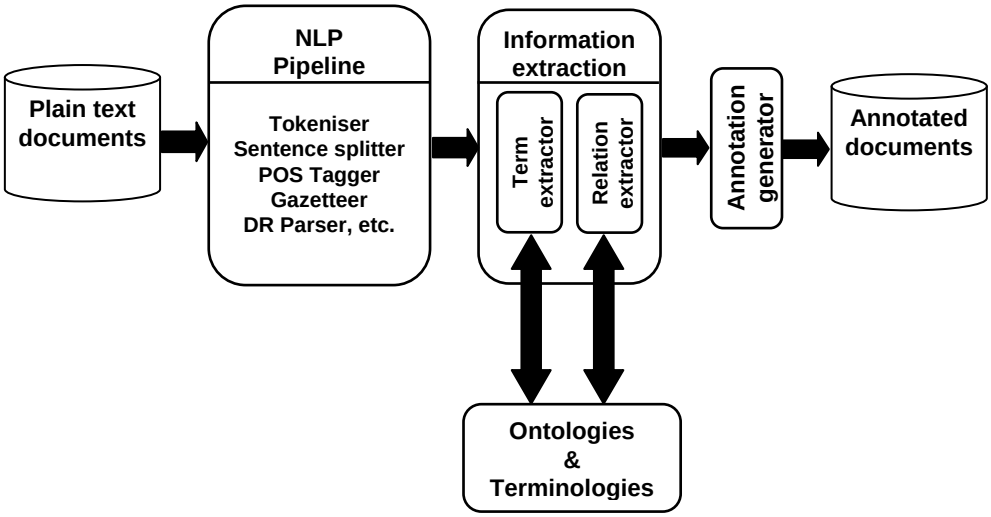

Figure 1. The dataflow of annotation generation from text

The **NLP pipeline** comprises different modules and techniques used to analyse texts (see section 3).
The **term extractor** module finds all their occurrences in the corpus using information produced by the NLP pipeline (see section 4.1) and ontology instances.
The **relation extractor** module is used to extract the relation instances that hold between terms. For this purpose, it uses information embedded in the ontology (the relationship hierarchy) and information produced by the NLP pipeline (see section 4.2).
The **annotation generator** collects information generated by all modules and generates a structured annotation based on the ontology. This annotation can be stored separately or embedded in the text document. In the semantic web context, most systems use the RDF language to represent these annotations.

3 The NLP pipeline: General techniques

The text analysis comprises several distinct stages beginning by breaking the text in words till the presentation of its contents. NLP systems implement either the totality of these stages, or a combination of certain stages.
The complete text analysis must go through the following steps:

- Morphological analysis: identification of word variations (plural form, abbreviation, etc.) and assigning some lexical information to each word (category, gender, number, etc.);
- Syntactic analysis: identifying the syntactic structures associated to each phrase (subject, verb object, etc.);
- Semantic analysis: building a set of semantic representations from the syntactic trees;
- Pragmatic analysis: identifying discourse items associated to each text.

These steps need the use of different techniques which include tokenisation, Pos tagging and parsing.

The **tokenisation** is the process of breaking the text into its constituent units called tokens (example of tool: Gate [10]). Tokens may vary in granularity depending on application but the most common method of tokenisation is the fragmentation of text into words and sentences (sentences splitting).
The **pos (part of speech) tagging** is the annotation of words with their appropriate pos tags taking into account their context within the sentence (example of tool: Treetagger [28]). The most common tags are: article, noun, verb, adjective, preposition, number, etc. Commonly, the pos tagging systems are based on rule taggers or on probabilistic models.
Parsing is the process of analyzing an input sequence in order to determine its grammatical structure with respect to a given formal grammar. In NLP, it allows to determine a complete syntactic structure of a sentence. For example, the output of a linguistic role parser is a tree, whose leaves correspond to individual words in the text, and whose nodes represent linguistic roles, such as Subject, Object, Verb (example of tool: RASP [5]), etc. a particular parsing, called *shallow parsing*, consists of computing word sequences or syntagms (phrases), which are a set of syntactically related words (example of tool: Syntex [4]). Each syntagm is then tagged by specific predefined tags, such as Noun syntagm, Verb syntagm, Adjective syntagm, etc.

4 Text mining in the biomedical domain

4.1 Term identification

A fundamental requirement in the biomedical text mining is the automatic identification of biomedical entities (terms) in the text. In the following subsections, we present some approaches proposed to facilitate this task.

4.1.1 Dictionary-based techniques

These methods use existing terminological resources (dictionary, lexicon, thesaurus…) with an aim of locating the occurrences of terms in the text. The application of the simple version of these methods, i.e. to find the direct correspondence between the entries of the dictionary and the textual entities does not give satisfactory results. These bad results are due primarily to problems of homonymy (i.e, in English for example, common words like `and', `by' or `for' are detected as gene names) and to problems of linguistic variations related to (i) the punctuation (mdm-2 and mdm2), (ii) the use of the Greek alphabet (p53alpha and p53α), and (iii) the word order (integrin alpha4 and alpha4 integrin).
In order to resolve these problems, much of improvements were added to these methods such as the use of dictionary of synonyms, the filtering of the stop words and the treatment of the variations. For example, in [19], the authors propose to code the dictionaries and the texts with the nucleic code (formed alphabet of 4 letters {A, C, G, T}) and to apply the BLAST algorithm [1] used for the alignment of DNA sequences, to identify the terms which have a strong similarity.

4.1.2 Rule-based techniques

These methods rely on the (manual) creation of extraction rules based on specific characteristics of a class of terms. These characteristics can be (i) morphological: the words ending in -ase and -in can be considered as enzymes or proteins and (ii) orthographical: the terms checking the regular expression [a-z] + [0-9] can be regarded as genes (a sequence of letters followed by a sequence of numbers). In [12], the authors propose a method for the automatic recognition of the protein names; they use the fact that the names of proteins are often in capital letters and comprise special characters and numbers. As for [14], an automatic tool of recognition of standard named entities (FASTUS [13]) is adapted for the recognition of the protein and gene names. This tool is based on finite cascade transducers. FASTUS allows to recognize complex units (for example: `3,4-dehydroproline'). A general rule-based methodology for the recognition of biomedical entities is presented in [2].

Other researchers use associations' rules which allow to highlight correlations between textual elements. A pre-processed corpus is used for the extraction of these rules which are then presented to a domain expert to validate them. Once validated, the association rules are classified according to probabilistic measurements and applied to the texts in order to extract terms. In [6], a methodology of biological text mining using the association rules is presented.

4.1.3 Machine learning-based techniques

As for all the methods based on machine learning algorithms, these methods use training data to learn features useful for term extraction and term classification. To each class, the algorithm assigns some orthographical characteristics (i.e combination of letters and numbers, term starting with a capital letter) or morpho-syntactic characteristics (extraction patterns). This information is then used by standard algorithms of classification which classify the terms in their adequate categories.
Several experiments were carried out by using various algorithms of classification, for example [9] is based on the hidden Markov models (HMM) whereas [15] uses the support vector machines (SVM). These methods are time- and resource-consuming; moreover, they are confronted to another problem which is the lack of corpus already processed to carry out their training. The majority of the experiments are carried out on the same corpus GENIA [18].
The Multi-field project CADERIGE [24] comprises several French teams of different competences (biology, training and NLP) with the aim of designing tools for machine learning text analysis. An editor of annotation was developed and a method of training of extraction patterns was developed.

Proux [26] and colleagues propose an hybrid approach where they combine different approaches. Their method uses a morphological pos-tagger which affects a special tag ("guessed") for terms that cannot be matched with classical transducers. So, gene names are tagged with this special tag and are confirmed through a contextual analysis or through the use of a dictionary.

4.2 Relation identification

The comprehension of biological, pharmacological or medical phenomena is generally based on the detection of an interaction between genes, proteins or molecules. Although a part of these interactions is stored in databases, a great part of them is expressed in natural language and thus stored in the biomedical publications. Several techniques aiming to extract these interactions have been reported in the literature:

- Techniques based on patterns or regular expressions generated by domain experts to extract biomedical entities connected by a relationship in the text.
- Techniques based on the learning of extraction rules form text and generalizing them to extract relations between terms.
- Statistical methods which predict relationship by looking for the co-occurrences of terms in the text.
- NLP-based methods which parse texts to find structured sentences from which relationships can be extracted.

For the detection of gene-gene and gene-protein interactions, in [23] the authors propose a method composed of three stages: (1) selection of the fragments of texts containing this kind of interactions, (2) the use of training algorithms on these fragments to define extraction rules, and (3) the application of these rules on the documents to extract the interactions.

In [29], a method to extract functional relations between genes is proposed. They consider that if two genes appear regularly in documents dealing with the same phenomenon (even separately), then a relation could exist between these two genes. They use statistical models which

describe the frequency of the words in the documents in order to classify them according to topics, and then deduce the functions of genes which appear in these documents.
The approach of Blaschke et al. [3] encapsulates representative relationships between proteins in common descriptions, called 'frames' and then evaluates the effectiveness of each frame in a large data collection. Examples of such frames are 'protein X binds to protein Y' and '…complex between protein A and protein B'. A comparable method was proposed by Ng et al. [25], it relies on a rule-based model with which they detect protein-activation or -inhibition relationships.

Several other approaches on the same topic are presented in [31] [30] [8]. The results of these systems make it possible to create networks of interaction between genes and proteins which can play an important role for example in the interpretation of experiments results or in the study of a particular phenomenon.

5 Some tools

5.1.1 Pubminer

Pubminer [11] combines techniques of training (HMM and SVM) with NLP techniques to process PubMed abstracts in order to extract named entities (gene, protein) and possible interactions between them. This system allows the visualisation of the results in a graph, where the nodes represent the names of genes and of proteins and the arcs represent the possible interactions; the user has also a link between the graph and the documents treated texts.

5.1.2 GeneWays

This system [27] allows (i) to select scientific journals, (ii) to identify gene/protein names in the journal text, (iii) to extract interactions between these genes/proteins and other actions by means of NLP and (iv) to store these interactions in a database.
GeneWays provides a Web interface that allows users to search and submit papers of interest for analysis.

5.1.3 Gis

GIS [7] is a text mining system based on decision tree approach. The sentence patterns in terms of wording and term distribution in describing relations are represented as a variant of decision tree. The system extracts relations classified into three categories - positive, cooperative and negative. It focuses on four types of gene-related information: biological functions, associated diseases, related genes and gene-gene interactions.

5.1.4 TextPresso

Textpresso [22] is an ontology-based information retrieval and extraction system for biological literature. It identifies terms (instances of the ontology concepts) by matching them against regular expressions and encloses them with xml tags. Textpresso also offers user interfaces to query these annotations. Annotation is embedded in the text, which makes difficult its reuse by other systems, and the term extraction phase needs thousands of regular expressions to extract relevant terms

5.1.5 MeatAnnot

Our tool, MeatAnnot, will be detailed in section 6.

6 Our approach

6.1 Motivations

As described below, the major need of biologists is to access knowledge described in natural language texts. Mining this literature is one way to detect relevant information and generate semantic annotations on documents in order to facilitate their search.
Our goal is to facilitate the information retrieval task for biologists. Therefore, in order to be able to create relevant annotations, we first tried to know in which information the biologist would be interested in an article. We thus studied how a biologist annotates a document: we provided three biologists with the same articles and asked them to annotate them manually.
This study revealed several common points between biologists' annotations, even if their ways of annotating were different. The information selected by the different biologists was almost the same. They primarily underlined the names of the studied genes, substances or proteins, the studied biological phenomenon or the cellular functions as well as the verbs describing a relation between these various elements.
An example of sentence annotated by the three biologists was: "KGF causes alveolar epithelial type II cell proliferation"; this sentence asserts that the substance KGF causes a type of cell proliferation.

The representation of this kind of annotation must be well defined, easy to understand by all biologists and unambiguous. To fulfill these requirements, this annotation should be based on a formal model of the biomedical domain (e.g. ontology).
The formalisation of the annotation scheme using the ontological hierarchy enables annotators to choose the appropriate level of annotation detail, helps to constrain the annotation structure, diminishes ambiguity and should reduce errors in the annotation process.
In addition, the fact that these annotations are based on ontology incites us to use standard formalisms such as RDF(S) [20] or OWL [21] which allow the reuse of these annotations by different annotation tools and search engines.
The approach chosen in this work is to reuse existing ontologies in order to support a text mining method applied on biomedical literature. These ontologies define the type of entities and relations that we aim to discover through text analysis and they allow generation of rich annotations about documents. These annotations can then be used to perform information retrieval task.

Figure 2 shows the different stages of this approach.

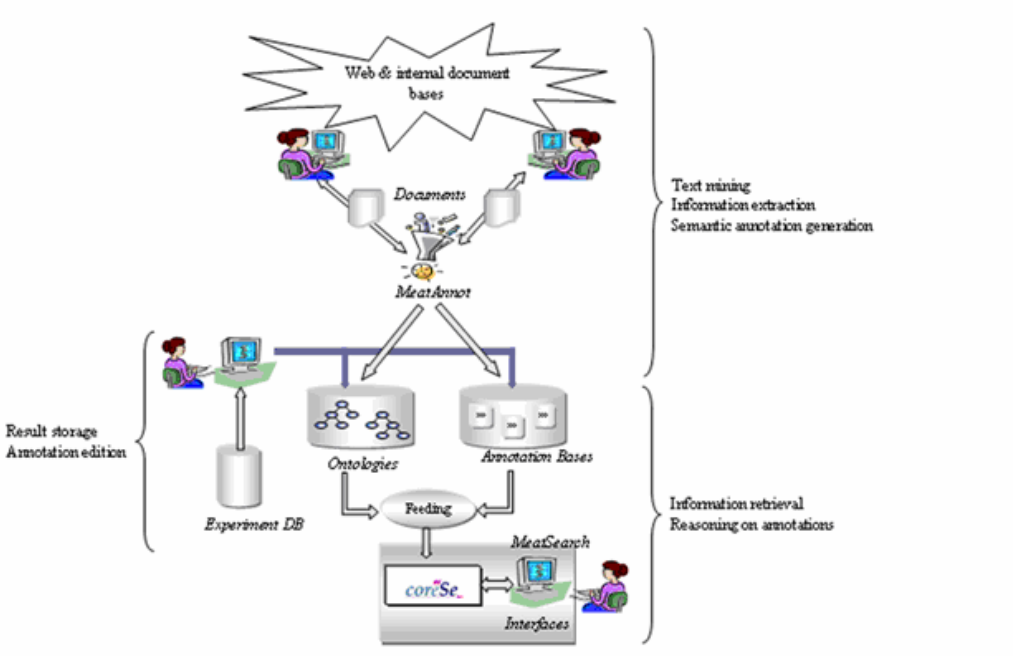

Figure 2. Ontology-based approach for generating and using semantic annotations

6.2 An ontology for the biomedical domain

Like in most research domains, biologists aim to represent, share and reuse their knowledge. Therefore, several terminological systems were proposed and developed: controlled vocabularies for annotating genes and indexing documents, thesauri for navigating among domain terms and for easing information retrieval. The success of these systems was limited because of their dependence on specific cases and tasks and the absence of reasoning.

As alternative for the limits of these resources, the biomedical community was interested in ontologies which aim at representing knowledge independently of any specific use[3]. Ontologies provide an organisational framework of the concepts and a system of hierarchical and associative relationships of the domain. In addition to the possibility of reuse and sharing allowed by ontologies, the formal structure coupled with the hierarchy of concepts and the hierarchy of relations between concepts offers the possibility to draw complex inferences and reasoning.

6.2.1 The UMLS project

The Unified Medical Language System (UMLS) project was initiated in 1986 by the U.S. National Library of Medicine (NLM). Its goal is to help health professionals and researchers to use biomedical information from a variety of different sources [32]. It consists of a (1) metathesaurus which collects millions of terms belonging to nomenclatures and terminologies defined in the biomedical domain and a (2) semantic network which consists of 135 semantic types and 54 relationships.

The semantic network represents a high-level abstraction for the metathesaurus; it is organized by distinguishing entities and events in two single-inheritance hierarchies. Each semantic type in the network has a textual definition and appears in one of these hierarchies.

The generation of ontology-based annotations on documents requires a lexicon of terms for referring to entities in the domain and an ontology describing this domain. For our case, this ontology must cover the entire biomedical domain (drugs, cells, genes, process…), but we no-

[3] We must notice that this independence on the ontology w.r.t. the application is strongly criticized by researchers such as the French TIA working group.

ticed that except UMLS, all other ontologies were developed for a specific case (for example, GALEN [3]is dedicated to clinical domain, MENELAS [44] to coronary diseases, GO [41] to molecular biology, etc.).
So, we chose the UMLS semantic network (SN) defined by [36] as upper-level ontology for the biomedical domain: the hierarchy of semantic types can be regarded as a hierarchy of concepts and the terms of the metathesaurus as instances of these concepts.
In addition, GO has recently been integrated into UMLS [33]. Overall, a total of 23% of the GO terms either match directly (3%) or are linked (20%) to existing UMLS concepts. All GO terms now have a corresponding UMLS concept. This integration offers an important link between medicine and genomics terms.

6.2.2 Enrichement of the relationships hierarchy

The UMLS semantic network comprises a hierarchy of 54 semantic relations made up of five families:

- *Physical relations:* connecting terms having common physic characteristic (example: branch_of);
- *Space relations:* connecting the terms according to their localisation (example: location_of);
- *Functional relations:* expressing a function or an activity connecting the terms (example: interacts_with);
- *Temporal relations:* connecting terms in time (example: precede);
- *Conceptual relations:* connecting terms according to some abstract concept, thought, or idea (example: measures).

After a thorough study of these various families and discussions with our colleagues biologists, it appeared that (i) to annotate a biological phenomenon, the two families primarily interesting for them are: conceptual relations and functional relations (65% of the whole set of the relations), and (ii) although these relations cover the totality of links that may exist between the concepts of the semantic network, some are too generic and can lead to a negative effect on the level of precision of an annotation.
For example, the functional relation ‘affects’ is defined by the production of a direct effect by a biological entity on another, this effect can be the result of the one of the following actions: has a role in, alters, influences, predisposes, catalyzes, stimulates, regulates, removes, pressure from, impedes, enhances, contributes to. This definition recommends that all these actions can be regarded as ‘synonymous’ and must be annotated by the relation ‘affects’. But this can generate noise in the annotations using this relation: for example, a biologist aiming to find all the biological entities stimulated by a particular gene, will have in addition to the correct entities, others which were deteriorated, catalysed... by this same gene.
Our goal is to use this ontology to annotate resources and to facilitate the task of information retrieval; therefore we decided to enrich the semantic network by more specific relations in order to have more precise annotations. So, we proceeded in two steps:

(1) Using relationship definitions
In this step, we relied on the definitions of each relation in the semantic network. As shown in figure 3, these definitions comprise a set of terms which can indicate a more precise sense of the concerned relation. These terms cannot be considered as synonyms of the concerned relation and some terms must be rather considered implicitly as more precise semantic relations. This assumption allowed us to specialise the UMLS relations by new relations. Figure 3 shows the result of this specialisation on the relation `affects'.

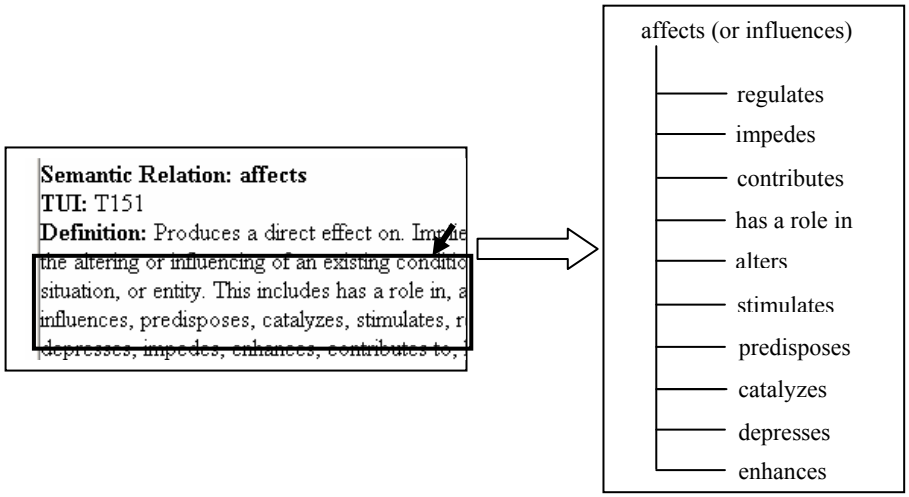

Figure 3. Example of relation enrichment: 'affects'

(2) Biologist's suggestions
During our discussions, the biologists proposed some relations specific to their field and which do not appear in the lists of terms characterizing the UMLS relations. This step enabled us for example to add the relation 'activates' and the relation 'inhibits' as two specialisations of the relation 'performs'. We thus succeeded in adding 24 new relations to the semantic network of UMLS. These new relations have the same signature as the relation to which they are attached.

Finally, we developed a script which translates each semantic type and each relation from its textual format towards the corresponding concept and property in RDFS language. Two semantic types (resp. relations) linked by an is-a link in the UMLS SN are translated into two RDFS classes linked by subClassOf (resp. subPropertyOf) property. In addition, we used some primitives of OWL Lite (restriction, cardinality, etc.) to solve some problems (discussed in [17, 34]) occurring in the definition of the signature (domain and range) of relations.

6.3 UMLS-based semantic annotation generation: MeatAnnot

6.3.1 Method

In spite of its advantages, the creation of semantic annotations is a difficult and time-consuming process for biologists. Therefore, we developed a system called MeatAnnot which, starting from a textual document (i.e. a scientific paper), allows to generate a structured annotation, based on UMLS SN, and describes the semantic content of this text.
MeatAnnot uses the NLP (Natural Language Processing) tools GATE [10], TreeTagger [28], RASP [5] and our own extensions dedicated to extraction of semantic relations and of UMLS concepts. It processes texts and extracts interactions between genes and other UMLS concepts.
So, for each sentence, it tries to detect an instance of an UMLS relation and to detect the instances of UMLS concepts linked by this relationship and it generates an annotation describing this interaction (see more details in [17]).
The generation method is decomposed in three steps described below:

6.3.1.1 Step1: Relation detection

In this step, we used JAPE [10], a language based on regular expressions and allowing us to write information extraction grammar for texts processed by GATE. So, for each UMLS relation (such as interacts_with, expressed_in, disrupts…), an extraction grammar was manually created to extract all instances of this relation.
The example below shows a grammar which allows detection of instances of the semantic relation "has_a_role_in" with its different lexical forms in the text (e.g. *has a role, had roles, plays a positive role,* etc.).
Example of grammar:

```
Rule:Has_role
 Priority: 1
(

 ({Tag.lemme == "have"}  |
  {Tag.lemme == "play"} )
  {SpaceToken}
 ({Tag.lemme == "a"}   |
  {Tag.lemme == "an"})
  {SpaceToken}
 ({Tag.cat == "JJ"}  {SpaceToken})?
  {Tag.lemme == "role"}
  {Tag.lemme=="in"})

  ):has_role -->
:has_role.RelationShip = {kind = "has_role", rule=Has_role}
```

In the above figure, Tag.lemme corresponds to the lemmatized form of the verb and Tag.cat corresponds to the grammatical category (JJ:adjective = important, vital, critical, etc.) of the term which can be present between the verb and the term 'role' ("?" means that it is optional).

6.3.1.2 Step2: Term extraction

To extract terms, MeatAnnot uses the Tokeniser module of GATE and the TreeTagger. The tokeniser splits text into tokens, such as numbers, punctuation and words, and the TreeTagger assigns a grammatical category (noun, verb...) to each token.
After tokenizing and tagging texts, MeatAnnot uses an extraction window of four (four successive words are considered as a candidate term) and for each candidate term, if it exists in UMLS, MeatAnnot processes the following word, otherwise it decreases the size of the window till zero.
To interrogate UMLS, MeatAnnot uses the UMLSKS (the UMLS Knowledge Server based on the MetaMap4 concept mapping program). This server provides access and navigation in the UMLS metathesaurus and in the UMLS semantic network. If the term exists in UMLS, the answer is obtained in XML format. This answer is parsed to obtain information about the term (semantic type, synonyms …); all this information is then used to generate the semantic annotation.
In this step we noticed that MeatAnnot cannot detect some gene names because of the increasing number of gene synonyms. To solve this problem, the biologists supplied us with a dictionary of specific genes used frequently in DNA experiments. So, after the extraction phase, MeatAnnot re-processes the text and tries to detect missing genes.
Some other specific biomedical terms were not detected by MeatAnnot (i.e. not found in UMLS).

[4] http://mmtx.nlm.nih.gov/

Example of sentence: "*ERK-5 also plays a role in the AP-1 regulation*"
In this sentence, MeatAnnot generates an annotation describing the relation (has_role_in) between the two genes *ERK-5* and *AP-1* since it cannot detect the term *AP-1 regulation* (since it does not exist in UMLS); this annotation is wrong or not relevant for biologists. Therefore, we developed some heuristics to solve this kind of problems:

H1: {term1.sty == 'Gene_or_Genome'}
{term2.string ∈ GF_terms} =>
{term3 = term1+term2; term3.sty = 'Genetic_Function'}

GF_terms ={'induction','translation','regulation','expression','mutation','deletion'}

H2: {term1.sty == 'Amino_acid_Peptide_or_Protein'}
{term2.string ∈ MF_termes } =>
{term3 = term1+term2; term3.sty = 'Molecular_Function'}

MF_termes ={'activity','binding','phosphorylation'}
….
H1 implies that, if a term detected as a gene instance is followed for example by the word "*regulation*", we can consider that the concatenation of both words is a 'Genetic_function' instance.
These heuristics can help to improve the term extraction phase and to enrich the UMLS metathesaurus with new terms and their associations to the SN.

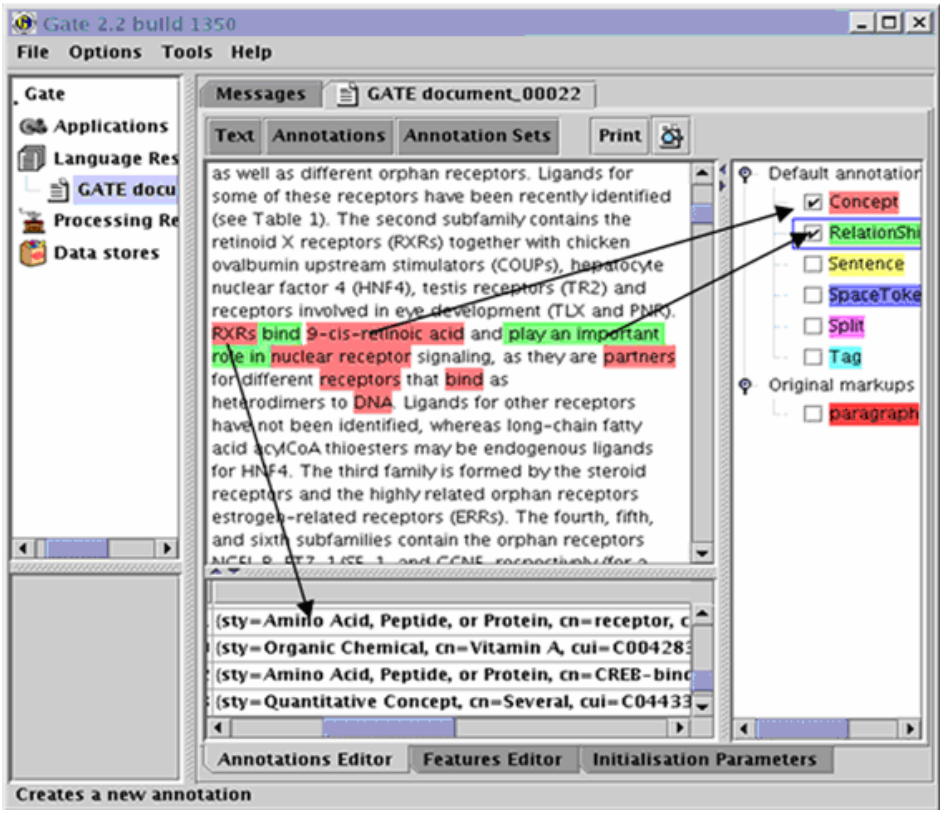

Figure 4. Example of relation detection and term extraction

Figure 4 is a GATE interface showing the obtained result after the two steps of relation detection and term extraction. In this example, two relations (bind and play_role) and seven terms were detected in this sentence.

6.3.1.3 Step3: Annotation generation

In this step, MeatAnnot uses the RASP module [10] which assigns a linguistic role (grammatical relation) to sentence words (subj, obj …): it allows finding out concept instances linked by the relation.
So for each detected relation, MeatAnnot analyses the results of the extraction phase and checks if the subjects and objects of this relation were detected as UMLS concepts. Then it generates an annotation describing an instance of this relation.
Since RASP processes only single words, the linguistic roles of multi-terms are deduced automatically by MeatAnnot. For example, in the sentence *"KGF causes lung injury",* RASP first assigns the object role to *"injury"* but MeatAnnot re-assigns this role to "lung injury" since it detected it as a UMLS concept instance.

The example below summarizes the process steps. Let us consider the sentence:
"IFN-alpha and IFN-beta are secreted by dendritic cells."

First: by applying the extraction grammars on this sentence, MeatAnnot detects (by the presence of verb 'to secrete') that it contains the UMLS relation "produce".

Second: Table 1 describes the result of the term extraction phase.

Table 1. Term extraction results

Term	Semantic type	Synonyms
IFN-alpha	Amino Acid Peptide or Protein	alfa-n3 interferon, Ginterferon, G-interferon, et.
IFN-beta	Amino Acid Peptide or Protein	Endogenous Interferon Beta, IFNb, IFN-B, etc.
dendritic cells	Cell	N/C

Third: MeatAnnot applies the RASP module on the sentence and parses the result to detect the different linguistic roles of the words.
An excerpt of the result of RASP on this sentence is:

```
(|ncsubj| |secrete+ed:5_VVN| |IFN-alpha:1_NN1| |obj|)
(|ncsubj| |secrete+ed:5_VVN| |IFN-beta:3_NN1| |obj|)
(|arg_mod| |by:6_II| |secrete+ed:5_VVN| |cell.+s:8_NN2| |subj|)
(|conj| _ |IFN-alpha:1_NN1| |IFN-beta:3_NN1|)
(|ncmod| _ |cell.+s:8_NN2| |dendritic:7_JJ|)
(|aux| _ |secrete+ed:5_VVN| |be+:4_VBR|)
```

Lines 1, 2 and 3 indicate that (i) the words "IFN-alpha", "IFN-beta" and "cells" are linked by the verb "secrete", and (ii) the linguistic role affected to IFN-alpha and IFN-beta (resp. cells) is object (resp. subject).

”dendritic cells produce IFN-alpha” and ”dendritic cells produce IFN-beta” are thus detected as instances of the relation “produce”; so, MeatAnnot generates an RDF annotation for these two instances and adds it to the annotation concerning this paper.

```
<m:Cell rdf:about='#dendritic_cells'>
    <m:produce >
      <m:Amino_Acid__Peptide__or_Protein  rdf:about='#IFN-alpha'/>
    </m:produce>
    <m:produce >
      <m:Amino_Acid__Peptide__or_Protein rdf:about='#IFN-beta'/>
    </m:produce >
</m:Cell >
```

After text processing, MeatAnnot generates an RDF annotation describing all these interactions described in the article and stores it in the directory containing the annotations of the other papers. Each article is linked to the RDF file containing its annotations. The current system has a flat annotation base; this base can be organized in the future, for example according to the article theme or to the user supplying with the corpus.
These annotations can then be used, either in a bibliographical search or in a more complex IR (Information Retrieval) scenario such as searching interactions between genes or of genes with other biomedical entities.

6.3.2 Towards a generic methodology

We presented a method based on semantic web technologies for generation of ontology-based semantic annotations for biological domain. This method can be generalized to any other scientific domains (chemical domain, physical domain, etc.) that have the same needs such as the support for the automatic generation of rich annotations from texts, which can help to validate and to interpret experimental results.
In fact, the modules presented are reusable and rely on standard technologies. The MeatAnnot method can be generalized in four points:

- Selection of an ontology covering the domain studied;
- Development of an API to query the ontology;
- Definition of the extraction grammar for the ontology relations;
- Reuse of MeatAnnot to detect terms and relationships and then generate semantic annotations.

Remark:
If the selected ontology needs enrichment or population by instances, it is possible to enrich it by using NLP tools on a textual corpus provided by the domain experts.
For example:

- Using Nomino[5] or Likes[6], to enrich and populate the concept hierarchy (as in the SAMOVAR system [47]).
- Using Syntex [4], to extract verb syntagms considered as relevant for the domain and which enable to enrich the relation hierarchy.

Figure 5 recapitulates the obtained method.

[5] http://www.ling.uqam.ca/nomino
[6] http://www-ensais.u-strasbg.fr/liia/likes/likes.htm

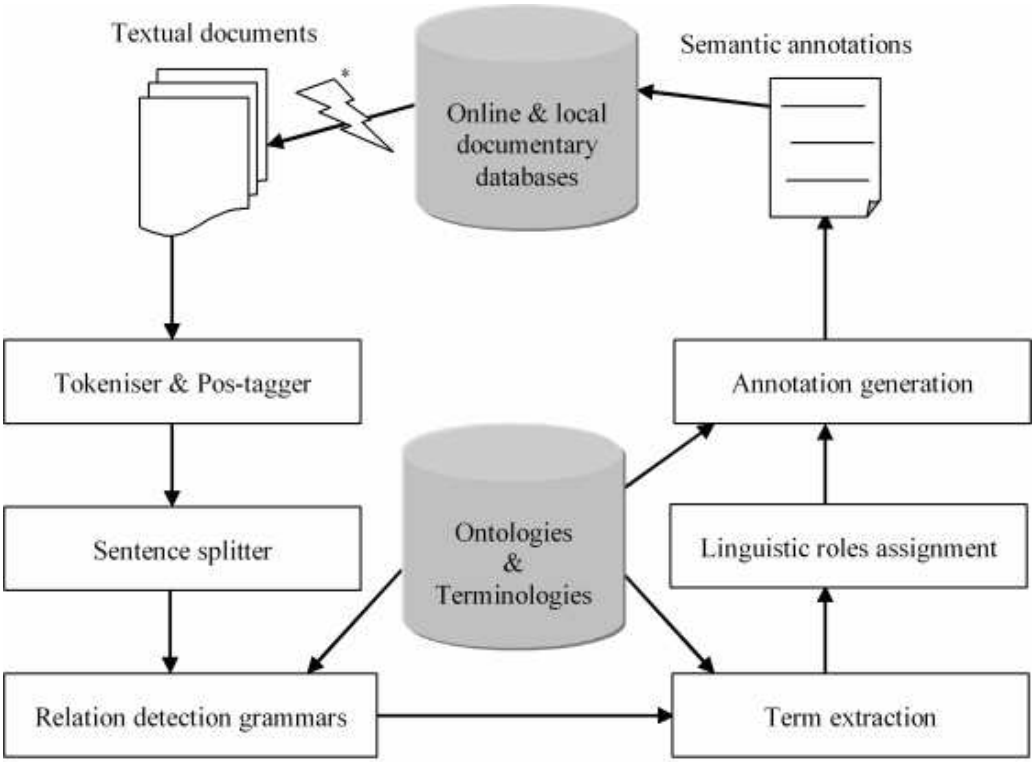

Figure 5. Ontology-based semantic annotation generation method

*: conversion of documents from their original format (generally PDF) to textual format.

This method allows the generation of semantic annotations based not only on concept instances but also on relation instances. In addition to document description, these annotations embed information about domain knowledge.

6.4 Evaluation of text mining techniques

6.4.1 Qualitative measurements

The quality of a text mining technique is typically assessed in terms of precision and recall.
Precision (P) relates to the absence of noise (also called commission) in the mining. It measures the ratio of "true positives" (terms or relations correctly identified) and both "true positives" and "false positives".
Recall (R) relates to the absence of silence (also called omission) in the mining. It measures the ratio of "true positives" and both "true positives" and "false negatives" (terms or relations that should be correctly identified).
The average is typically measured by a single measurement called **F-score** (or F-mesure), which calculates the harmonic value of precision and recall.

$$F-score = 2 \times \frac{P \times R}{P+R}$$

In [16], we proposed a new measurement called **usefulness** to measure the ratio of suggestions considered as useful for the biologists and correct suggestions. In their work, not useful sugges-

tions are ontology relationship instances relating concepts instances and describing a basic or vague knowledge for the biologist.
This measure is subjective since it relates to a point of view of a user or of a group. In this work, the annotations considered as useless by a biologist are stored in the annotation base. A possible improvement would be to add metadata on these annotations (for exemple: useless_for) which would allow to filter the answers sent by this biologist.

These different measurements give an idea about the performance of a particular technique, but to compare different techniques, we must take into account several characteristics other than the performance.
Here are some characteristics we consider important to compare different techniques:

- **Type of text**: it concerns the kind of the text which we want to analyze: abstracts vs. full-text articles vs. database fields vs. informal documents (such as mails).
- **Input format:** it concerns the text formats accepted as input. The majority of text mining tools takes plain text as input, so, taking into account others format such as PDF, HTML, etc. can be considered as advantage.
- **Output format:** it concerns the result format. The use of standard format (text Unicode, XML, RDF, etc.) is preferred to facilitate the reuse of results by other applications.
- **External resources:** it concerns the resources used to support the text mining method such as dictionaries, ontologies, thesauri, etc. The quality and the availability of these resources can have an effect on the technique performance. In addition, ontologies provide computable semantics which can help to improve the text mining technique (disambiguation, predefined relations, restriction, etc.).
- **Automation:** it concerns the degree of automation of the text mining technique (semi-automatic, automatic). Aiming to help users in the annotation task, fully automated techniques are preferred when voluminous amount of documents need to be annotated.
- **Type of analysis:** it concerns the model used by the technique to analyze text (machine learning model, statistical model, fully NLP model). This characteristic can give an idea about the consumption of time and of resources by the analysis.
- **Type of result:** it concerns the kind of the identified entities. In the biomedical domain for example, results can be: only gene or protein names, different biomedical entities, relations between biomedical entities, instances of ontology concepts and relations, etc. Techniques which provide the maximum of information are preferred.

Table 2 presents the comparison of text mining techniques implemented by tools presented in the section 4.3.

Table 2. Comparison of text mining tools using our defined characteristics

	TextPresso	**Gis**	**Geneways**	**Pubminer**	**MeatAnnot**
Type of text	Full-text articles	Medline abstracts	Full-text articles	Medline abstracts	Full-text articles
Input format	HTML, plain text	HTML, plain text	Plain text	Plain text	Plain text
Output format	XML	N/C	Database	N/C	XML, RDF
External resources	Own ontology	Lexicons	Knowledge base	UMLS metathesaurus	UMLS semantic network and metathesaurus
Automation	Automatic (definition of regular expressions)	Automatic	Automatic (with lerning phase)	Automatic (with learning phase)	Automatic
Type of analysis	Pattern matching (definition of regular expressions)	Decision tree	Machine learning	SVM and HMM (machine learning)	NLP (definition of relation identification grammars)
Type of result	Concepts instances and some interactions	Gene-related information (biological functions, disease, etc.)	Interaction between molecular substances (genes, proteins, etc.)	Genes, proteins, interaction between them	UMLS concepts instances related by UMLS relations instances

6.4.2 Evaluation of MeatAnnot:

To validate our annotations, we adopted a user-centred approach: we chose randomly a test corpus (2751 sentences) from the documents given by biologists and we presented the suggestions proposed by MeatAnnot to biologists via an interface in order to evaluate their quality.
Since these annotations were intended for an IR context, we focused on classic IR quality measures for indexing and we adapted them to our case.
We noticed also that some suggestions were considered as correct but not useful to the biologists since they described a basic or vague knowledge. Therefore, we introduced a new measure, called usefulness, for measuring the rate of useful suggestions.
This measure is subjective because it relates to a point of view of a user or of a group. In this work, the annotations considered as useless by a biologist are stored in the annotations base. A possible improvement would be to add metadata on these annotations (for exemple: useless_for) which would allow to filter the answers sent by this biologist.

Table 3. Definition of measurements

	Measures
Precision	Nb suggestions correctly extracted / Nb all suggestions extracted
Recall	Nb suggestions correctly extracted / Nb suggestions that should be extracted
Usefulness	Nb useful suggestions extracted / Nb suggestions correctly extracted

Precision relates to the absence of noise (also called commission) in the extraction and recall relates to the absence of silence (also called omission).

Table 4. Results:

	Suggestion	Correct	Missing	Useful	Precision	Recall	Usefulness
Result	509	426	274	399	0.836	0.608	0.936

The second column describes the number of relations correctly extracted from texts. The difference with the number of suggestions proposed by MeatAnnot is mainly due to the errors generated by the NLP tools (e.g. wrong grammatical category or wrong linguistic role) and to the terms missing in UMLS (i.e. when the subject or object of a relation was not found in UMLS). Nonetheless a good precision is obtained since 83% of the suggestions were correct.
The third column describes the number of relations not extracted by MeatAnnot: these missing suggestions are also due to the errors generated by the NLP tools and mainly to relations deduced by the biologist (when s/he reads the sentence) and which cannot be generated automatically.

Example of errors generated by the NLP tools:

"TRP gene, which belongs to the TRP-homolog group, is expressed in neurons"
In this sentence where the relation "expressed_in" is detected, the RASP module suggests that "which" is the subject of the relation, so MeatAnnot does not generate the annotation because "which" is not an UMLS term and it loses the interaction between the "TRP gene" and "neurons".

Example of missing relations:
"Upon interferon-gamma induction, after viral infection for example, a regulator of the proteasome, PA28 plays a role in antigen processing."
In this example, MeatAnnot extracts automatically the relation "PA28 plays_role antigen processing" but a biologist who reads this sentence can deduce, using his/her implicit knowledge, another relation which is "interferon-gamma have_effect PA28".

Finally, MeatAnnot has a good usefulness since 93% of correct suggestions are considered as useful by biologists. The annotations regarded as useless are however added to the RDF file containing the other annotations: they have no negative impact and they may be relevant to novice or non expert users.
These results prove that MeatAnnot generates good quality annotations, an essential feature for a use in an information retrieval context.

7 Conclusions

7.1 Discussion

This document presents a methodology for the generation of ontology-based annotation by extracting information from texts; it surveys different text mining techniques proposed for the biomedical domain and it presents an approach to compare these techniques.
Building a whole application to generate automatically ontology-based semantic annotations presents some difficulties:

- The choice of the best text mining technique adapted to our needs.
- The choice of one or several ontologies covering the biomedical domain (biological entities, diseases, medical entities…).
- Integration of the different components presented in Figure 1 (different output/input, different programming languages, etc.).
- The conversion of articles generally from PDF format towards plain text.

We also presented an approach based on semantic web technologies for generation and use of ontology-based semantic annotations. The generated semantic annotations can be used in several scenarios, such as:

- Improvement of the document retrieval phase: the use of the concepts/relations hierarchies to expand users' queries improves recall.
- Discovering new knowledge: the CORESE semantic search engine we use for exploiting the semantic annotations generated by MeatAnnot can find out paths between two entities. A path is constituted by a set of relations. In our case, biologists can deduce the role played by a selected gene in a disease by analysing the path found out between them (such a technique was used in semantic web service aggregation [39]).

Another originality of our work consists of the use of several technologies to provide a real world Corporate Semantic Web Application that (i) relies on formal semantics (Ontologies, Semantic annotations) which reduce ambiguity compared to informal semantics, (ii) offers drawing inferences on these semantics at runtime (by using CORESE), and (iii) uses text to extract information (NLP tools) which is a very rich source of knowledge.

Last, we think that an evaluation study on the generated annotations (as the evaluation proposed in this paper) is necessary since this generation phase is expensive and often irreversible.

Finally, this paper proposes some solutions to problems raised during the final discussion of W3C Workshop on Semantic Web for Life Sciences [46]:

- Good quality of the annotations extracted automatically: MeatAnnot annotations.
- Adequate representation of the context: our metadata on annotations which gives new ways of reasoning and more information on the annotation base.
- Possibility of reasoning on annotations: CORESE enables such reasoning.
- Semantic web browsing: we offer an automatic association of semantics to the knowledge resources and we provide a user-interface support.

7.2 Lessons Learnt and Further Work

We can distinguish several kinds of lessons learnt: from conceptual and methodological viewpoint, from technical viewpoint and from applicative viewpoint.

Technical lessons learnt:
In this work, we tested and used several NLP tools for building our information extraction system. The first problem raised in this phase was the component integration; this problem is due to the difference between the input/output formats of the different tools. We solved this problem by using the GATE API which (i) provides tools such as tokeniser, pos-tagger, gazetteer… and (ii) offers the possibility to integrate any new component (existing tools such as RASP, TreeTagger or our own extensions). Moreover, the conversion of articles from PDF format towards plain text is very problematic for several reasons: (1) PDFs are generated by several different tools, (2) biologists often use Greek alphabet characters that are difficult to recognize, (3) journals and books have different layouts for article presentation. For this phase conversion, we used an OCR (Optical Character Recognition) software; it gives good results but it requires user's intervention. So, it is necessary to develop an automatic converter taking into account all these problems.

Performance evaluation:
We adopted a full text analysis approach while most of existing systems process only article abstracts. Our approach is clearly time-consuming (the complete processing of a page takes about 3 minutes) but it is worthwhile since it increases the recall/precision of the information extraction phase and gives more information about knowledge embedded in texts. The installation of a local version of UMLS and some technical optimisations can decrease the running time of the system. Moreover, as this phase is a batch preprocessing independent of the later real-time processing of any user query, it prepares more efficient query processing.

Discussion on ontology reuse:
Our approach based on UMLS confirms that reusing existing domain ontologies can help to build real-world semantic web applications. In fact, despite some knowledge engineering problems (discussed above), the use of an existing upper-level ontology (such as UMLS Semantic Network) coupled with a rich terminology (UMLS metathesaurus) facilitates the information extraction process and allows to generate rich and shareable annotations. We think that the UMLS-based approach should be generalized to different domains needing interpretation and reasoning.
Several researchers have emitted doubts about possible reuse of ontology. They insist on the influence of the intended application on the ontology: some modeling choices or some ontology structuring choices are influenced by the future application aimed. But our experiment in

MEAT project clearly showed the interest of using UMLS as reference ontology. In our work, the ontology was altogether the reference w.r.t. to which the annotations were created, the terms extracted and the relations extracted. To confirm the interest of such reference ontologies, for relations, we had first relied on Syntex tool that offered both term extraction and verbal syntagms extraction from a corpus of sample articles in biology. Our objective was to propose an extraction grammar for each relation indirectly expressed by these verbal syntagms (independently of any reference ontology, so as to offer a bottom-up approach, and to rely only on relations attested by texts). But it appeared that all the interesting relations were already included in UMLS relations. It confirms that a library of relation extraction grammars (written in JAPE) can be reusable by other researchers aiming at extracting UMLS relations from biomedical texts. The validation phase can be useful for indicating that some relations were not adequately extracted because of the lack of accuracy of the grammar for extracting this relation. So this phase can enable to refine this relation extraction grammar. A tool such as Syntex can be useful in this purpose.
The good results obtained in the information extraction phase confirm that the automation of this task is useful and it eases the user's work. In addition, the use of standard semantic web technologies for formalisation of this information into ontology-based semantic annotation can solve the knowledge sharing problem.
Even more, we think that this method can be adopted to annotate online documentary databases (such as Pubmed); the annotation base obtained might represent a very rich knowledge source for biologists. Nevertheless, we must not forget that an assumption underlying annotation generation by MeatAnnot is the consistency of the different articles and the absence of contradictions among them. This hypothesis enables to reason about the global RDF annotation base containing all the annotations stemming from the different articles, as in a global knowledge base. Therefore if we tackle the whole Pubmed articles, we must be watchful about the coherence of the obtained annotations since contradictory biologist's viewpoints or wrong results may be contained in these articles.

Discussion on W3C standards:
By generating RDF annotations, we rely on W3C recommendations for semantic Web. For queries and rules, there is not yet any official recommendation. However, a W3C working group works on SPARQL as future query language recommendation. CORESE query language – that we used in MEAT - is very close to SPARQL and handles most SPARQL features. Moreover, CORESE has the advantage to already offer processing of queries expressed in this language. Moreover it must be noticed that the MEAT end-users – biologists – use user interfaces for expressing their queries and do not directly handle this query language: these internal queries are generated automatically from the user interfaces. Concerning the rules, so far, there is not yet any rule language recommended by W3C. Therefore the best solution was to use CORESE rule language for which CORESE offers a rule engine that has been working quite efficiently since several years [48].

Further work:
In future versions of MeatAnnot, we aim to take into account contextual information during the knowledge extraction phase. Let us take the example of the following sentence: "In vitro assays demonstrated that only p38alpha and p38beta are inhibited by csaids". Actually, in this sentence, MeatAnnot identifies that 'p38alpha' and 'p38beta' are inhibited by 'csaids' but does not detect the fact that this inhibition is observed 'in vitro' whereas this information can be very important for the interpretation of a particular result.
MeatSearch can also be improved by introducing some typical search scenarios proposed by biologists.

Finally, like for most semantic web applications, we must propose solution to manage the ontology evolution. In fact, such evolution can cause inconsistencies in the annotation base, which induce errors in the information retrieval phase.

Acknowledgements
We thank Remy Bars from Bayer Cropscience, the IPMC[7] team working on microarray experiments, especially Kevin Le Brigand for fruitful discussions and PACA[8] region which cofunds this work by regional grant.

8 Bibliography

[1] Altschul S., Boguski M., Gish W. and Wootton J. – Issues in searching molecular sequence databases – *Nature Genetics 6:119-129*, 1994.

[2] Ananiadou, S. – A Methodology for Automatic Term Recognition – *Proceedings of COLING-94. Kyoto, Japan. p. 1034-1038*. 1994.

[3] Blaschke, C. et al. – The frame-based module of the Suiseki information extraction system – *IEEE Intell. Syst. 17, 14–20,* 2000.

[4] Bourigault D., Fabre C., Frérot C., Jacques M.-P. and Ozdowska S. – Syntex, analyseur syntaxique de corpus – *Actes des 12èmes journées sur le Traitement Automatique des Langues Naturelles, France*, 2005.

[5] Briscoe E. and Carroll J. – Robust accurate statistical annotation of general text – *Proceedings of the Third IC LR E, Las Palmas, Gran Canaria. 1499-1504*, 2002.

[6] Cherfi H., Napoli A. and Toussaint Y. – Towards a Text Mining Methodology Using Association Rules Extraction – *Soft Computing Journal*, 2005.

[7] Chiang, J.-H., Yu, H.-C., Hsu, H.-J. – GIS: a biomedical text-mining system for gene information discovery – *Bioinformatics, 20, 120–121*, 2004

[8] Cohen AM and Hersh WR – A survey of current work in biomedical text mining – *Briefings in Bioinformatics. 6: 57-71*, 2005.

[9] Collier N., Nobata C., and Tsujii J. – Extracting the Names of Genes and Gene Products with a Hidden Markov Model – *Proceeding of COLING 2000, pages 201–207*, 2000.

[10] Cunningham H., Maynard D., Bontcheva K. and Tablan V. – GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications – *ACL'02*, 2002.

[11] Eom J.-H. and Zhang B.-T. – PubMiner: Machine learning-based text mining system for biomedical information mining – *Proceeding of the 11th International Conference on Artificial Intelligence: Methodology, Systems, and Applications (AIMSA), Varna, Bulgaria*, 2004

[12] Fukuda K., Tsunoda T., Tamura A., and Takagi T. – Toward information extraction: identifying protein names from biological papers – *PSB, pages 705–716*, 1998.

[13] Hobbs, J.R., Appelt D., Bear J., Israel D., Kameyama M., Stickel M. – FASTUS: A Cascaded Finite-State Transducer for Extracting Information From Natural-Language Text – *Finite-State Language Processing., Cambridge: MIT press. 383-406*, 1997.

[14] Hobbs, J.R. – Information extraction from biomedical text – *Journal Biomedical Informatics. In Proceedings of Pac Symposium Biocomputers. p. 541-552*, 2000.

[15] Kazama J., Makino T., Ohta Y., et Tsujii J. – Tuning svm for biomedical named entity recognition – *Proceedings of the workshop on NLP in the biomedical domain*, 2002

[16] Khelif K., Dieng-Kuntz R., Barbry P. – Semantic web technologies for interpreting DNA microarray analyses: the MEAT system – *Proceeding of WISE'05, 20-22/11 New York*, 2005.

[7] http://www.ipmc.cnrs.fr
[8] http://www.cr-paca.fr/

[17] Khelif K. – Web sémantique et mémoire d'expériences pour l'analyse du transcriptome – *Phd thesis, Nice Sophia Antipolis Univeristy,* 2006.

[18] Kim, J.D., Ohta, T., Tateisi, Y., and Tsujii, J. – GENIA corpus - semantically annotated corpus for bio-textmining – *Bioinformatics 19(Suppl. 1), i180-182*, 2003.

[19] Krauthammer M., Rzhetsky A., Morozov P. et Friedman C. – Using BLAST for identifying gene and protein names in journal articles – *Gene 259(1-2):245-52*, 2000.

[20] Lassila O. and Swick R. (2001) – W3C Resource Description framework (RDF) Model and Syntax Specification – *http://www.w3.org/TR/REC-rdf-syntax/*, 2001.

[21] McGuinness D.L. et Van Harmelen F. – OWL Web Ontology Language Overview – *http://www.w3.org/TR/owl-features/*, 2004.

[22] Muller H.M., Kenny E.E. and Sternberg P.W. – Textpresso: an ontology-based information retrieval and extraction system for biological literature – *PLoS Biologie, E309*, 2004.

[23] Nédellec C., et al. – Machine learning for information extraction in genomics state of the art and perspectives – *Sirmakessis, S. (ed.) : Text Mining and its Applications. Studies in Fuzzi. and Soft Comp. 138. Springer Verlag, Berlin Heidelberg New York 99-118*, 2004.

[24] Nédellec C. and Nazarenko A. – Application de l'apprentissage à la recherche et à l'extraction d'information - Un exemple, le projet Caderige : identification d'interactions géniques – *Actes de la Journée thématique Exploration de données issues d'Internet, Bennani Y., et al. (Eds)*, 2001.

[25] Ng S.K. and Wong M. – Toward routine automatic pathway discovery from on-line scientific text abstracts – *Genome Inform. Ser. Workshop Genome Inform. 10, 104–112,* 1999.

[26] Proux, D., F. Rechenmann, L. Julliard, V.V. Pillet, and B. Jacq. – Detecting Gene Symbols and Names in Biological Texts: A First Step toward Pertinent Information Extraction – *Proceedings of Ninth Workshop on Genome Informatics p. 72-80*, 1998.

[27] Rzhetsky A., Iossifov I., Koike T., Krauthammer M., Kra P., Morris M., Yu H., Duboue PA., Weng W., Wilbur WJ., et al. – GeneWays: a system for extracting, analyzing, visualizing, and integrating molecular pathway data – *J Biomed Inform, 37:43-53*, 2004.

[28] Schmid H. – Probabilistic part-of-speech tagging using decision trees – Proceedings of the International Conference on New Methods in Language Processing. Manchester, 1994.

[29] Shatkay H., Edwards S. and Boguski M –Information retrieval meets gene analysis – IEEE Intelligent System (Special Issue on Intelligent Systems in Biology). 17:45-53, 2002.

[30] Shatkay H. and Feldman R. – Mining the biomedical literature in the genomic era: an overview – Journal of Computational Biology, 10, 821–855, 2003.

[31] Staab S, editor – Mining information for functional genomics – IEEE Intelligent System 17-66, 2002.

[32] Humphreys B. and Lindberg D. – The UMLS project: making the conceptual connection between users and the information they need – *Bulletin of the Medical Library Association 81(2): 170*, 1993.

[33] Lomax J. and McCray A. – Mapping the Gene Ontology into the Unified Medical Language System – *Comparative and Functional Genomics, 5:354–361*, 2004.

[34] Kashyap V., Borgida A. – Representing the UMLS Semantic Network using OWL: (Or "What's in a Semantic Web link?") – *In ISWC'2003. Heidelberg: Springer-Verlag; 1-16*, 2003.

[35] Humphreys B. and Lindberg D. – The UMLS project: making the conceptual connection between users and the information they need – *Bulletin of the Medical Library Association 81(2): 170*, 1993.

[36] McCray A. – An upper level ontology for the biomedical domain – *Comp Functional Genomics; 4: 80-84*, 2003.

[37] Rector A., Rogers J.E. and Pole P. – The GALEN High Level Ontology – *Fourteenth International Congress of the European Federation for Medical Informatics, MIE96, Copenhagen, Denmark,* 1996.

[38] Nédellec C. – Bibliographical Information Extraction in Genomics – *IEEE Intelligent Systems & their Applications, p.76-80*, 2002.

[39] Gandon F., Lo M., Corby O. and Dieng R. – Managing enterprise applications as dynamic resources in corporate semantic webs: an application scenario for semantic web services – *In W3C Workshop on Frameworks for Semantics in Web Service, http://www.w3.org/2005/04/FSWS/,* 2005.

[40] Ohlbach H.J. and Schaffert S. – *eds Workshop on Principles and Practice of Semantic Web Reasoning at the 20th ICLP, St Malo, France,* 2004.

[41] Ashburner M., Ball C., Blake J., Butler H., Cherry J., Corradi J., Dolinski K., Janan T., Eppig T. and Harris M. – Creating the Gene Ontology resource: design and implementation – *Genome Res, 1425–1433,* 2001.

[42] Vargas-Vera M., Motta E., Domingue J., Lanzoni M., Stutt A. and Ciravegna F. (2002) – MnM: Ontology Driven Semi-Automatic and Automatic Support for Semantic Markup – *In Gomez-Perez A. and Benjamins R. eds, Proc. of the 13th International Conference on Knowledge Engineering and Knowledge Management EKAW 2002, Springer Verlag LNAI 2473, 379-391,* 2002.

[43] Séguéla P. and Aussenac-Gilles N. – Extraction de relations sémantiques entre termes et enrichissement de modèles du domaine – *In IC'99, Paris, 79-88,* 1999.

[44] Zweigenbaum P. – MENELAS: an access system for medical records using natural language – *Comput Methods Programs Biomed. Oct;45(1-2):117-20,* 1994.

[45] Handschuh S., Staab S. and Ciravegna F. – S-CREAM – Semi-automatic CREAtion of Metadata – *The 13th International Conference on Knowledge Engineering and Management (EKAW 2002), ed Gomez-Perez, A., Springer Verlag,* 2002.

[46] Summary Report – W3C Workshop on Semantic Web for Life Sciences. – http://www.w3.org/2004/10/swls-workshop-report.html, 2004.

[47] Golebiowska J., Dieng-Kuntz R., Corby O. and Mousseau D. – Building and Exploiting Ontologies for an Automobile Project Memory – *First International Conference on Knowledge Capture (K-CAP), Victoria, October 23–24,* 2001.

[48] Corby O., Dieng-Kuntz R. and Faron-Zucker C. – Querying the Semantic Web with the CORESE engine. In R. Lopez de Mantaras and L. Saitta eds – *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI'2004), Valencia, Spain, IOS Press, p.705-709,* 2004.