From Linguistic to Conceptual: A Framework Based on a Pipeline for Building Ontologies from Texts

Ali Benafia[1], Smaine Mazouzi[2], Ramdane Maamri[3], Zaidi Sahnoun[3], Sara Benafia[4]
[1]University of Batna, Batna (Algeria)
[2]University of Skikda, Skikda (Algeria)
[3]University of Constantine, Constantine (Algeria)
[4]Enaf of Batna , Batna (Algeria)

ABSTRACT: *This paper presents a novel approach of information extraction for building ontologies covering an extensive range of applications from corpora. Our goal is to propose a method that is independent of domains and based on a distributional analysis of semantic units to bring out all the candidates informative elements (concepts, entities, semantic relations, named entities ...).This method is based on a pipeline of four main stages allowing to refine the extraction information from unstructured text in the form of a suite of decomposable representations (sentences in triplets, 'argumental structure'…) until to get a consistent final ontology.*

We applied the pipeline defined in the context of a repeated sampling of 100 articles randomly drawn from text corpus ('Le Monde' with annual version '2013'). For the evaluation results of the trial implementation of our system , we have achieved a level of accuracy at which was up to 74% . We believe from the results obtained that our methodology is quite generic, and can be easily adapted to any new domain.

Keywords: Ontology, Information Extraction, Text Analysis, Similarity Measure, Linguistic Processing, Terminology Recognition

Received: 8 June 2017, Revised 17 July 2017, Accepted 26 July 2017

1. Introduction

Construction of the ontology depends strongly on information extracted from the various data sources. In the manual approach, the analysts of the domain base themselves on classical techniques to collect information, such as discussions with the experts of the domain or by manual consultation of documents. However, such a process is extremely expensive in time and resources and also raises productivity and quality. In the semi-automatic approach, ontology building using texts relies often on the process of text analysis, whether it is according to statistical approach [1]: The natural language processing tools are used to analyze texts and extract semantic concepts and relations. In fact, a text is an important source of information and knowledge, which is constant and shared by different communities. Texts contain linguistic elements such as entities named ,terms, semantic classes, relations…, which are very useful to the to the ontology building of the domain.

In addition, texts are more available than experts of the domain that intervene at the modeling level. It has, however, to be noted

that the construction process cannot be fully automatic, because the results of extractors are noisy, and that necessitates a subjective judgment of the ontology expert [2]. To be able to extract information from text, many approaches of automatic processing of natural languages exist in the literature. Yet, there is no standard methodology which constitutes a framework more consistent for building of the ontology using text expressed in natural language.

In general, the best known approach is consisted of six main stages used in methodologies of ontology construction using text. These stages [28] are :**a)** identification of the purpose and scope of the ontology , **b)** constitution of documents corpus , **c)** depth linguistic analysis of the corpus and normalization , **d)** specification and formalization of the ontology , **e)** ontology reuse and finally **f)** evaluation of ontology by analyzing requirements specification and competency questions.

When ontology is built, it can describe objects and put them into context (e.g., people, places, events, relationships, etc.). Reasoning systems rely on ontologies to provide extensive formal semantics that enable the systems to draw complex conclusions. In contrast, systems that extract information from unstructured sources as the text use much lighter-weight ontologies to encode their results, because those systems are generally not designed to enable complex reasoning. Ontologies have been applied to a number of different domains, including biomedicine , finance, tourism , education , natural language processing and software engineering .

In this paper, we propose a novel methodology for the automatic construction of ontology from any large corpus. This methodology, based by pipeline, encompasses several models and algorithms that can be used and combined in order to the construction of ontology.

The remainder of the paper is as follows. In Section 2, we will discuss the related works in this field. In Section 3, we will give our methodology and an overview of the overall system architecture concerning ontology building. The description of the preprocessing chain, the thematic segmentation of texts and the information extraction are presented in more detail in sections 4, 5 and 6. In Sections 7, 8, 9 and 10, we present our model for the building of extended ontology thus that the experimental results then we compare our approach with other approaches and conclude and mention our future work.

2. Related Works

Ontology construction from texts has been widely studied these last years. This process provides integrated pipelines for automatically computing concepts, relations between concepts and inference rules from text analysis, either linguistic or statistical. In this line of research, [3] presents a domain-independent method for automatically learning terms from the Web for the building of ontologies. This kind of approach has a substantial impact on performance of ontology construction. However, [3] showed that these indicators vary greatly for one relationship to another, but also for a corpus to another for same relationship. Another approach is offered by [15] to enrich systems question / answer or text summary: it aims to identify semantic relations between named entities. This method involves to bring out of homogeneous classes of named entities pairs, each class shall be considered as representative of an interesting relationship for the domain considered. Despite his interest, this approach is limited by the fact that it is based solely on named entities.

In [26], we find a method for extending ontology tree using natural language processing .This method is considered as effective solution for unstructured data extraction. It allows extracting new concepts from Web and linking the new concepts with the concepts in Yahoo based on a clearly defined relationship. [17] have presented a semi-automatic ontology extension using spreading activation. This approach allows to identify hierarchical relationships such as subsumption, head noun analysis and WordNet consultation are used to confirm and classify the found relationships.

[31] have proposed a semi-automatic approach based on the user-interactive dialogue system for knowledge acquisition, where, the user is engaged in a natural-language mixed-initiative dialogue. The system Sofie [23] extracts ontological facts from text corpus and can link them to ontology. For extraction of facts and relations, the system achieves high accuracy grace to logical rules obtained manually.

In [11], the authors have presented an approach that extracts ontology from text documents by Singular Value Decomposition (SVD), which is a pure statistical analysis method, as compared to heuristic and rule based methods. They adopt Latent Semantic Indexing (LSI), which attempts to catch term-term statistical references by replacing the document space with lower dimensional concept space. Their method is convinced of its simplicity but limited with precision.

3. Proposed Approach

Generally speaking, there are four common stages in all methodologies of construction and enrichment of domain ontology by using text:1) constitution of corpus , 2) linguistic analysis of corpus , 3) linguistic analysis of corpus and 4) formalization of the ontology.

The overview of our approach is illustrated in the figure against. In general, it starts with a textual corpus and an initial ontology, and after four processing stages, it outputs a set of simple mappings.

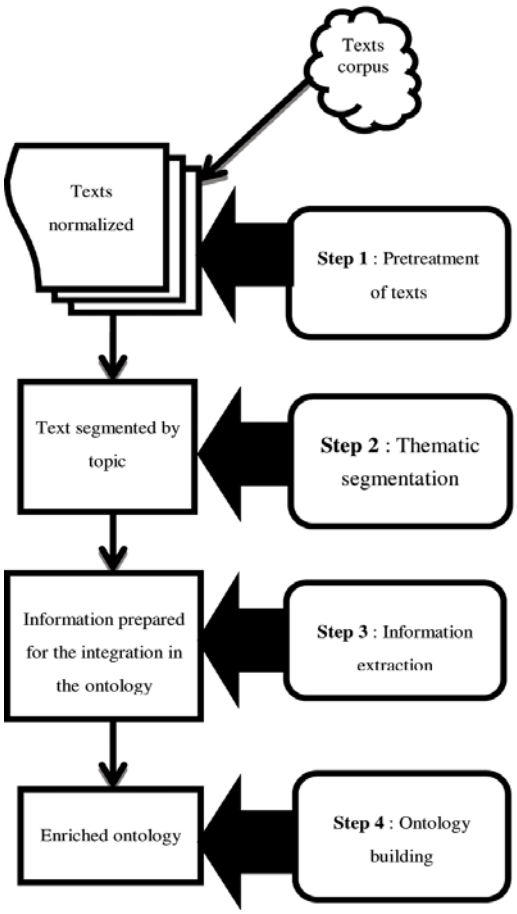


Figure 1. An overview of the proposed approach to building extended ontology

Our approach is different from the others approaches encountered in the literature and comprises a pipeline containing four phases that are:

Phase 1: Pretreatment of texts allowing to eliminate the possible ambiguous we may encounter in texts.
This process is important for the subsequent treatments.

Phase 2: Thematic segmentation, which consists in identifying the most important thematic segments in a text in order to cut it into homogeneous passages .This task is essential for extract the set of close terms in the texts.

Phase 3: Information extraction for analyzing unrestricted text in order to extract information about prespecified types of events, entities or relationships.

Phase 4: Definition of an adequate ontology to model diversified domains and their construction. We will describe these steps in more detail in the following sections.

4. Description of the Preprocessing Chain

Text preprocessing is usually known as the task of transforming a raw text file, essentially a sequence of characters, into a well-defined sequence of linguistically meaningful units [21]. Text pre-processing is an important part of our approach, since the characters, words, and sentences identified at this procedure are the substantial units passed to all further processing stages. In our case, the proposed text preprocessing is roughly divided into the following phases: tokenization, normalization, identification terms and part-of-Speech Tagging.

4.1 Tokenization

This is the process of splitting a text into individual words or sequences of words (n-grams). It makes a first segmentation non-linguistic segmentation. The token is considered as the basic textual unit. This kind of segmentation follows the recommendations of TC37SC4/TEI group [22]. We consider in our case four kinds of tokens alphanumeric (alpha, alpha capital), numeric (0, 1…), separators (punctuation marks) and symbols (hyphen, delimiters...). The result of this procedure is a set of words which needs to be used in the following treatment.

4.2 Normalization

This process permits to recognize the named entities (numbers, places, dates …) in the text. The words belonging to named entities should be labeled, denoting that in later processing phases these elements are to be handled in a special way, e.g. the morphology analyzer will not analyze these kinds of words .The role of this task is to avoid any ambiguities that may exist in the sentences. For this, we used an analysis tool of "entities named": TagEN [9], this tool has been tested on several applications and gives very good results.

4.3 Identification Terms

A term can have multiple forms (simple word, compound word or a complex word as collocations). To identify separately each form, we found it useful to assign each case a priority (the collocations then the compound words and finally the other remaining words). To do this, we perform the proposed treatment in three passes:

- First passage: Identification of Collocations

We identify the possible collocates of each word by parsing the text snippets returned by the search engine when querying that word. Then, we rank the list of syntactic co-occurrences retrieved according to the collocational strength of each pair by using a set statistical measures [6].

- Second passage: Identification of Compound Words

We have addressed this problem and have proposed a new heuristic to search for compound words in a text [7]. -*Third passage*: treatment of remaining words.

The remaining words were stored as a "*bag of words*", which is a representation of text as an unordered collection of terms that disregards word order or grammar. The remaining words that are not collocations, compound words or named entities, will be considered as unknown words. A specific treatment was affected for this kind of words [6].

4.4 Part-of-Speech Tagging

In our case study, we used the TreeTagger analyzer [13] where lemmatization process is performed simultaneously with the morpho-syntactic process where lemmatization process is performed simultaneously with the morphosyntactic process.

4.5 Dependence Structure between Terms

It allows to create a representation for each sentence defining dependencies between different words. The tool chosen for this type of treatment is Syntex analysis [13].

5. Thematic Segmentation

The thematic text segmentation task consists in identifying the most important thematic parts in a text in order to cut it into homogeneous passages.

5.1 The Concept of Theme

The theme concept is difficult to define and the specialists in linguistic that attempted to characterize it have given many definitions. We will take in this context, a definition which seems to us most interesting, that presented in [10]:" the notion of topic is clearly an intuitively satisfactory way of describing the unifying principle which makes one stretch of discourse about something and the next stretch about something". A second definition is given in the same paper: "For any text, there is a single correct phrase which is the topic, and titles are a number of different ways of expressing the topic" .Related to this issue, Tyler [27] also declares that there are many possible titles for any text.

The topic can only be one possible paraphrase of a series of utterances. We will consider these two definitions in our problem of thematic segmentation.

5.2 Measures used for Segmentation Thematic

The object of our study is first of all to determine thematic segmentation of the text to highlight all nearby segments and thus to obtain linguistic sub-schemas that we obtain by integrating them into our ontology.

Our approach, here, uses the principle of lexical cohesion. It is based on an original heuristic performing a series of recursive evaluation using a set of scores related to well-defined criteria. We present below these scores.

5.2.1 Frequency Characteristics of a Term

TF-IDF is a way to score the importance of words (or "terms") in a segment based on how frequently they appear across multiple segments (corpus).

$Score_{Tf\text{-}idfs}(t) = tf_s(t)*log(N/df_s(t))$ where $tf_s(t)$ is the number is the number of occurrences of the term t in segment s, $df_s(t)$ is the total number of segments wherein the term is present and N the total number of segments.

Note: when the measure $Score_{Tf\text{-}idfs}$ of candidate term is high then the presence of this element will be important in analyzed segment…

5.2.2 Co-occurrence Factor

We use for this kind of score, the measure of mutual information on the co-occurrence of two terms, the term t with a term taken among the others terms t_i of segment S.

We have then $IM(t, t_i)=log2(N*a\ (t, t_i))/((a\ (t, t_i)+b\ (t_i)*(a\ (t, t_i)+d\ (t)))$ where N : total number of co-occurrence in the text; a : number of co-occurrence between the term t et t_i ; b : number of co-occurrences between the term t_i and all other terms of the text except the term t_i and c : number of co-occurrences between the term t and all other terms of the text except the term t.

The final score is then: $Score_{coccurrence}(t,s) = \sum_{i=1}^{n} IM\ (t, t_i)$ with n, the number of terms in the segment s.

Note: Argument of the element is different from arguments elements b and c.

5.2.3 Semantic Proximity between Terms

The notion of semantic proximity considered here allows to evaluate the distributional similarity of term t with other terms t_i present in the analyzed segment.

In general, the semantic similarity between word pairs is modeled by their related concepts[31]. Several approaches are lexical resources based among these approaches some are dictionary based , some are thesaurus based and other are Wordnet based .In our study, we choose resource Wolf [23] considered as a lexical database gigantic where synsets are interlinked by means of conceptual-semantic . We estimate the semantic proximity between two terms based on the depth in wolf and that of their least subsumed.

Each term or concept $T\text{-}Ci$ is represented by a vector $Vi = (LL_{i1}, LL_{i2}, LL_{i3},.. LL_{ij},\dots LL_{im})$ with LL_{ij} the relationship level meaning with others words or concepts and m the number of components of concept vector Vi.

We define LL_{ij} as follows:

$$LLij => \begin{cases} \text{Level of node corresponding to T-}C_j \text{ in the tree of concepts} \\ \text{Level of node corresponding to } T\text{-}C_j \text{ in the tree of concepts if } i = j \\ 0 \text{ otherwise} \end{cases}$$

The similarity between two terms $T\text{-}C_i$ et $T\text{-}C_j$ is obtained by the formula: $Sim(T\text{-}C_i , T\text{-}C_j) = (V_i * V_j)/(|V_i| * |V_j|)$

Algorithm 1. Score Calculation Algorithm

Input: *t*, term to compare with the other terms t_i of segment *s*

A, tree of concepts subsumed belonging to the resource Wolf with each concept possess a semantic field.

Output: $score_{semantic\text{-}proximity}$, cumulative score for the term t with the all other terms t_i of segment *s*

1: $score_{semantic\text{-}proximity} \leftarrow 0$

2: Locate *t* in the concept tree *A* and let *c* the corresponding node

3: For each term t_i of *s* do

4: Begin

5: Locate t_i in the concept tree and let c_i the corresponding node

6: $score_{semantic\text{-}proximity} \leftarrow score_{semantic\text{-}proximity} + (V * V_i)/(|V| * |V_i|)$; { in the general case ,the similitude for two vectors V_i and V_j is defined by the formula: $Sim(T\text{-}C_i, T\text{-}C_j) = (V_i * V_j)/(|V_i| * |V_j|)$}

7: endfor

Note: *V* and V_i denote the vectors of concepts corresponding to terms *t* and t_i

5.2.4 Centrality Score Target '*term* '/ '*terms of segments*'

Local centrality of a concept in a segment is expressed by the combination of the above scores. As the first score is of statistic type and the last two are of linguistic type then the overall score expressing the centrality of the term *t* in a segment *S* is given by:

$$Score_{centrality}(t, S) = \alpha * (Score_{Tf\text{-}idfs}(t, S) + Score_{coccurrence}(t, S)) + (1 - \alpha) * (Score_{semantic\text{-}proximity}(t, S))$$

Where α ($\alpha \in [0, 1]$) is a weighting factor which allows to balance the frequency of occurrence (a statistical contribution) compared to the relevance of the words (linguistic contribution). In our case, this parameter is defined empirically to **0, 43**.

5.3 Proposed Heuristic

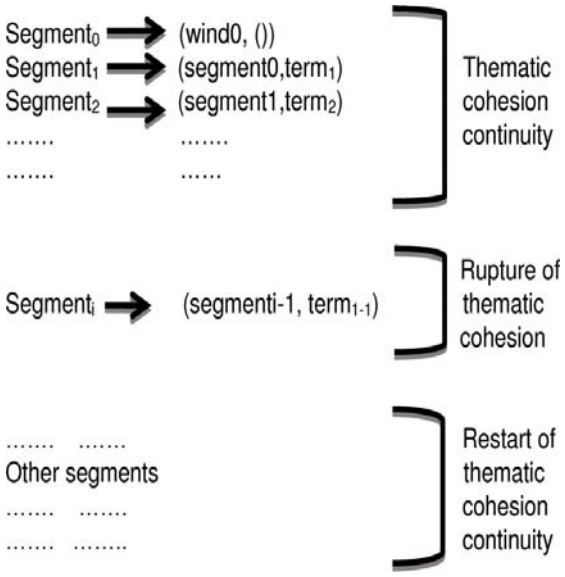
The rupture of thematic cohesion at segment-level *I* generate a new window *Wi* of size sz and starting from the term $term_{i-1}$.We compare the term $term_i$ with the all terms of window *Wi* and the process continues recursively until the exhaustion of terms of the window *Wi*.

Algorithm 2. Detect of Rupture of Hematic Cohesion

Input: *W*0 , the initial window ; *sz* its size ; *t*0 is the first term following the set of terms situated in the window W0; tolerated_threshold , is a threshold chosen which gives the relative margin tolerated in the topic segmentation of texts.

Output: *segm*, the segment thus constructed.

1: segm $\leftarrow \emptyset$



2: take a window *W*0 of size *sz*

3: score ← 0

4: *W*1⟵*W*0 U {*t*0}

5: *score1*⟵*score* (*W1*)

6: *if score1-score*0 < *tolerated_threshold then*

7: Begin {in this case there is rupture of thematic cohesion in the sequence of segments s}

8: *segm*⟵*segm U* {*W*0}

9: Get new window *W*0 {*the* window *W*0 is acquired from the term *t*0 and its size is equal 2}

10: end

11: else {there is always thematic cohesion and in this case we keep we keep the term t0 in the window and we progress iteratively of a term}

12: *W*0⟵*W*0 U {*t*0}

13: end if

14: take a term, from text analyzed, in *t*0

15: Resume the action (4) until there are more elements in the text

Note: The last term in the text is indicated by a special mark

5.4 Merge of Close Segments

This process of merge is carried out using a combination of features associated at segment. These features are chosen such that they are relevant to the description of the segment structure.

5.4.1 Description of Segment Features

The features of each segment consist of ten features which are summarized in the following table:

-Position of beginning of the segment in text.

-Nearest-neighbor distribution terms of segment.

-Weighting of terms in segment.

-Concept attributed to the topic segment.

-Named entities (number of named entities for " PLACE", "DATE", of "PERSON"…).

-Number of words in common with other segments.

-Number of synonymous terms with other segments.

-Number of co-occurrences of segment with all other segments.

-Number of terms in distributional relationships with other segments.

-Lexical coverage rate with the vocabulary of text.

5.4.2 KNN Learning for Merging Segment

Learning adopted here is based on the set of instances in which there is no explicit description of the function to learn.

In our case, this function is replaced by a set of attributes describing the notion of "thematic cohesion". When a new segment arrives in the base of examples, this segment is compared with all examples of the base in function to attributes defined (10 in number , see section 5.4.1). We determine the k closest examples, each of these *k* examples votes so that the segment is stored in its own class and the class that received the highest votes "wins" and will be selected. Hence the chosen class can be expressed as:

$$ArgMaxi \sum_{j=1}^{k} sim(Sj, S) * \delta (C (Sj), i)$$

Where $sim(S_j, S)$ is the similarity between the segment S_j (example segment) and the new segment S ; $C(S_j)$ is the class of segment Sj and δ the Kroneker function that's equal to 1 if both arguments are equal, 0 if not. We choose the class C_i whose vote is equal to $ArgMax_i$. At the end of every segment of each class processes will be merged into a single segment containing same thematic specifications.

The thematic text segmentation task consists in identifying the most important thematic parts in a text in order to cut it into homogeneous passages.

6. Information Extraction

6.1 Linguistic Resources

We have three linguistics resources:

***-Base of examples*:** it is obtained from a corpus composed of newspaper articles on various topics taken from the French daily '*Le Monde*' issued in annual version 2013 .

***-Base of examples in the form triplets*:** The sentences obtained in this normalization procedure became grammatically simple and do not contain nested proposals. They are built around the substantial element which is the verb (cf 6.2).

***-FrameNet* (in French):** We propose to take advantage of this semantic database for building a database containing the morpho-syntactic patterns in language change. This resource is accessible directly on the Web at: "*http://www.experts-exchange.com/Networking/Misc/Q_21967521.html*".

6.2 Decomposition of Sentences in Triplets

The position of the verb in the sentence allows to split this phrase into two parts, the left part and the right part. We proceed by cutting each sentence around the verb, this decoupage gives two syntagms: a left syntagm and another right and we continue recursively this process of decoupage while the obtained syntagms contain always a verb. The stopping criterion of this recursive process of decoupage into noun syntagms (left and right) stops once the examined syntagm no longer contains the verb.

Taking the Following Example:

This image contains a large house located on edge of the river.

This sentence is supposed to be already labeled by TreeTagger and Syntex. The decomposition procedure described above turns this specification in form (triplet):

< This image, ***contains***, a large house located on edge of the river >.

We apply the same process again for the right part of the triplet thus generated; this syntagm is not completely nominal which gives two triplets:

< This image, ***contains***, a large house >
< A large house, ***located***, on edge of the river >

The process stops because all the syntagms in the left and right of the triplets are nominal. These two triplets obtained are interrelated.

6.3 Acquisition of Lexico-syntactic Patterns

This process transforms the segments (structured triplets) to a representation in the form of a set of syntactic patterns in in agreement with the verb.

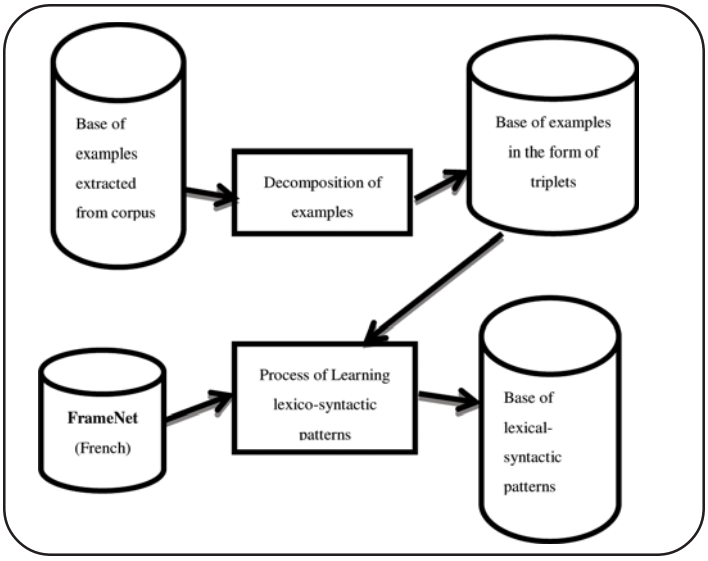


Figure 2. Schema of acquisition of lexical-syntactic patterns

For the extraction of the kind of patterns, we have developed a learning heuristic from Wikipedia corpus that we describe as follows:

Algorithm 3. Construction of base of lexico-syntactic patterns

Input: Base of examples in the form of triplets; FrameNet (French)

Output: Base of lexico-syntactic patterns {the set of lexical-syntactic patterns obtained will serve us as a model to perform the quantification of the structure below}.

1: Apply on the corpus considered, the rules of recognition of objects involved in the target relation for each verb phrase.

2: Extract of the corpus all the sentences containing related objects in the target relation.

3: Select manually the sentences in which the target relation is located between the pair of objects corresponding to the target relationship.

4: Repeat this process between each couple of sentences from the set of previously selected sentences.

5: For each pair of selected phrases do

6: Calculate their minimum distance.

7: Extract the most representative pattern in term of generalization.

8: end for

For the calculation of the minimum distance between sentences that can be coupled, we applied the algorithm for determining the optimal alignment between two sequences of strings [18].

Finally, this cover patterns-based is deemed sufficient, the reason they gave is that our ontology is oriented towards modeling management applications (is a part of, contain, define, compose ...).

6.4 A Generic Model Named 'Argumental Structure' and Instantiation of its Support

6.4.1 Definition of Model

The semantic and understanding of texts is a theory initiated by Fillmore in the late 60's. In [14], the author postulates that we can not understand the meaning of words and their arrangement in an optimal way unless we take into account by taking the event or situational context in which it is located. The argument classes retained for the integration of semantic roles for the sentences of corpus are eight in number. These arguments are defined in the table 1 and modeled come indicated in the figure below.

Type of argument	Description
What	Specific information in other terms – CHARACTER, OCCUPATION, etc., of a PERSON– true nature or identity of something or the sum of its characteristics – reason or purpose of something…
Who	What person or persons – what character, origin, position, importance, etc. – person that or any person that used relatively to represent a specified or implied antecedent– ask a question about the identity …
Whose	To identify a specific agent (PERSON, …) observed in proposition...
Where	In or at what place, position … – in what position or circumstances…
When	At what time or period – how long ago…
How	In what way or manner, by what means – to what extent, degree, etc. – in what state or condition…
Why	For what? For what reason, cause, or purpose – for which , on account of which (usually after reason to introduce a relative clause)…
How_many	Request for specific information– to what extent or degree, how much…

Table 1. Classes of model '**Argumental structure**

6.4.2 Justification for the choice of arguments for our model

According to Hanks [16] , the senses are built around the verb, the pivot of proposals. In [5] the authors consider that, the verbs have a tendency to be rather phraseology i.e. the values of a specialized verb are mostly determined by other lexical items around it. It is impossible to know the meaning of some verbs without considering the phraseology context or lexical-syntactic environment where they are.

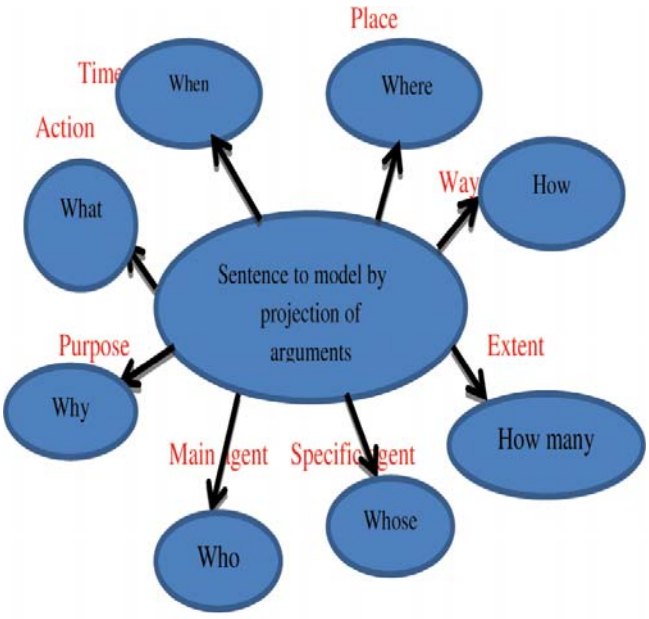


Figure 3. Modeling of arguments

Any sentence is qualified as a projection of generalized sentence that includes all the arguments presented above.

These arguments are then sufficient to determine the semantic roles associated with each sentence in the corpus considered.

6.4.3 Assigning Semantic Roles

The goal of this part is to describe the chain that allows us to pass of text segment to structure named "A*rgumental structure"* defined in Section 6.4. The latter structure is used directly for building ontology.

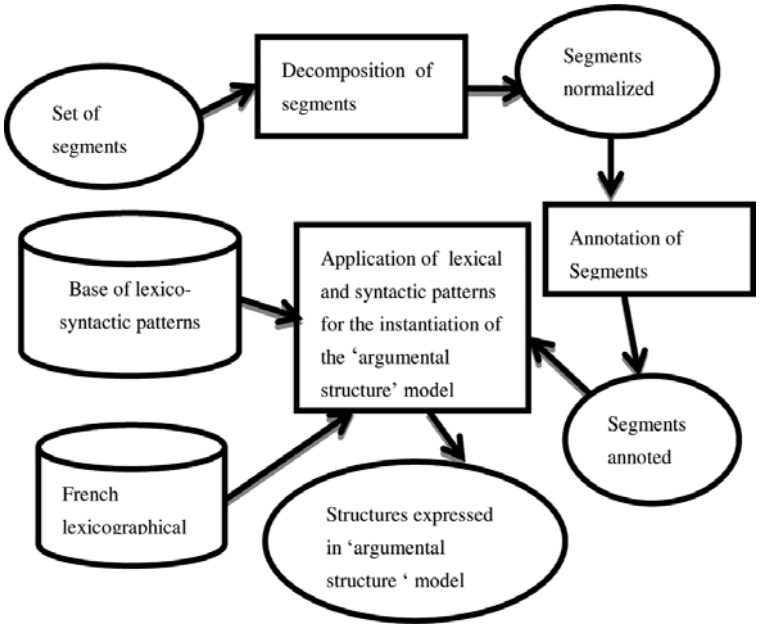


Figure 4. Instanciation of 'Argumental structure ' model

6.4.3.1 Annotation Triplets

In the lexical-syntactic patterns base, we find, for each verb, a set of lexical and syntactic forms that describe all possible lexico-syntactic configurations with other categories that can match him .With this base, we also have another resource (French lexicographical) that combine classes of verbs in syntactic semantics.

For example, the verb "lead" is considered here as the substantial element of the sentence, his presence in the patterns base allows us firstly to give all possible lexico-syntactic configurations and also its presence in the resource French lexicographical will provide us the syntactic-semantic properties. By matching the morphological structure of the sentence to deal with structures generated by the verb, we get a more refined form to be used for instantiating the class structure of arguments.

Algorithm 4. Annotation Triplets

Input: triplet t^i_k of segment s_k, base of lexico-syntactic patterns, French lexicographical resource.

Output: triplet t^i_k of segment s_k annoted in semantic roles

1: **for** each triplet t^i_k associed to segment s_k **do**

2: identify the verb *v*

3: annotate (left part of the triplet t^i_k,*v*)

4: annotate (right part of the triplet t^i_k, *v*)

5: **end for**

Procedure Annotate (Part Left or Right of triplet t^i_k, *v*)

1: Identify the actants according with linguistic rules using labels of syntactic dependencies determined by the verb v located in left or right part of t^i_k.

2: Transform the structure obtained by syntactic parties using the lexical-syntactic base.

3: Search in the dictionary of verbs, the classes related to the verb v

4: **for** each class identified **do**

5: **for** each pattern of identified class **do**

6: Decompose the pattern in arguments

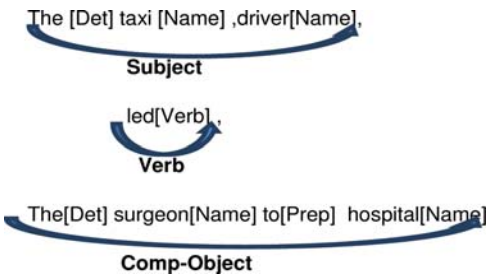
7: Place the corresponding roles in the triplet structure

8: **end for**

9: **end for**

To clarify this process, we consider the following example:
"*The taxi driver led the surgeon to hospital*"

Before the normalization of the sentence, all terms have already been segmented and identified (named entities, simple words, compound words ...), labeled by TreeTageer [13] and structured in dependencies terms by Syntex [13], giving the following form:



The procedure of normalization allows to normalize the sentence in the form of triplets from the verb "led" where we will have: (The taxi driver, led, the surgeon to hospital)

By applying the above algorithm, we get:

-subject [PERSON] verb led object [PERSON] preposition to [HABITATION] (sense1)

-subject [PERSON] verb led object [PERSON] preposition to [PERSON] (sense2)

-subject [PERSON] verb led object [PERSON] preposition to [ETAT PSYCHOLOGIC] (sense3)

We have three patterns: the first two correspond to the same sense of verb led, operate a vehicle (sense 1) and to compel to act in a particular way (sense 2). Focusing on the first two results, and patterns respectively, we see that they describe the same type of event – a person being accompanied by car to a certain location. In fact, both hospital and human are understood as geographical points where the respective entities are located. We retain this structure:

Subject [the taxi driver: Person] verb [led: Action] objet [the surgeon: Person] preposition [to] [the hospital: Habitation].

6.4.3.2 Arguments Instantiation

This treatment applies to triplets (syntagm left part, verb syntagm right part) with or without the verb (verb part may be empty). In the structure obtained previously, the syntagm left part is the actant while the syntagm right part defines the circumstants.

-The verb corresponds always to an argument WHAT; it represents the predicate of the action (e.g. WHAT: = "led").

-The subject corresponds to an actant that is to the main agent responsible for carrying out the action, the associated argument is WHO e.g. (WHO: = "taxi driver ").

-WHOSE is an element which achieves the action is indicated by the substantive (e.g. WHOSE := "surgeon").

-A second substantive is added to the previous indicating the place WHERE (e.g. WHERE: = "hospital").

-As to the other arguments, they are absent in the analyzed structure (WHEN: = "NULL", WHO: = "NULL").

7. Model for the Building of Extended Ontology

7.1 Description of Our Ontology

Today, there are a large number of ontologies available on the web. The use of these ontologies depends to the kind of requested task, however, for a complex task, several heterogeneous ontologies seem necessary for a better management of the problem. Our ontology is open to multi-task and provides sufficient flexibility to handle a wide variety of applications such as indexing, text comprehension...

Our ontology includes the static aspect (F2, F5), the dynamic aspect (F1, F4) .It also includes structuring objects in relational form with concept Lattices (F2) which can help to determine the generalization and specialization of each object.

7.1.1 Intention of Ontology

The intention of the ontology in our case is the union of different frame's instances that constitute the ontology.

Let O be ontology to describe and F_i the ith frame in the composition of the ontology .We have then: $I(O) = U^m{}_1 F_i$) with m , the number of frames .Each instance F_i is defined as quintuple: (V_i, E_i, N_i, T_i, C_i) where:

V_i is a set of structure in the form of verbs defining the dynamic of the ontology evolution .These verbs are extracted from text corpus with all the terms in association.

E_i is a set of structure of objects related to verbs. Each object is descripting by these features.

N_i is a set of structure of named entities located in text corpus. Each named entity is a fully specified reference and is marked with its type of class to which it belongs (PERSON, ORGANIZATION...).

T_i is a set of structure of verbs with its arguments which specified the potential thematic roles corresponding. This structure plays a major role in multiple ways.

C_i is a set of structure of constraints which contains the valid configurations of reality. They are expressed as rules, axioms, terms, formulas…

7.1.2 Extension of Ontology

Each them topici is described in our ontology by a frame framei composed of a set of slots representing the concept of segment where a slot is divided into five facets $f_1, f_2 \ldots f_5$. The frame structure is organized as follows:

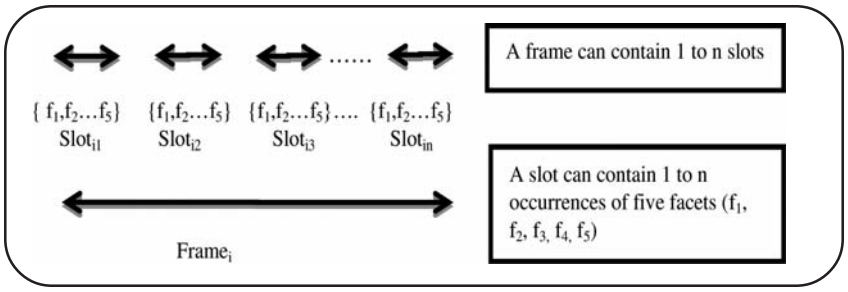


Figure 5. Frame extension of our ontology model

Formally the facet structure F_i (i = 1, 5) has the following form:

{Link} {Objects} {Named entities}*{Linguistic attachment} * {constraints} *

where Link, Objects, Named entities, Linguistic attachment and constraints contain respectively the values of V_i, E_i, N_i, T_i, and C_i.

The problem is to integrate $f^i{}_k$ with the elements of $f^r{}_i$

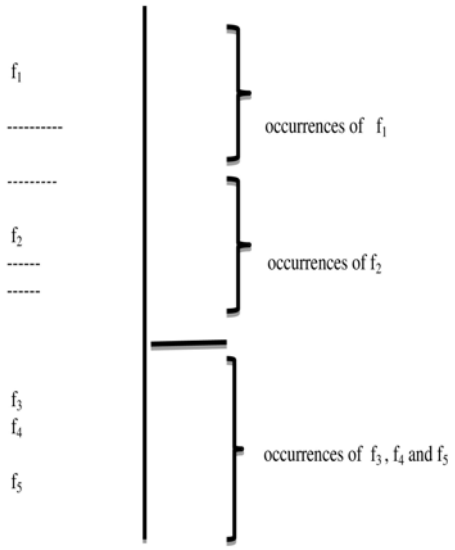


Figure 6. Facet extension

7.2 Automatic Construction and Enrichment of Ontology

We consider three main steps:

- The first concerns the processing of the "Argumental structure " on ontological structure.

- The second determines from the base of triplet features (cf table. 2) the element closest to the structure obtained in the previous step.

- The Third adds the element thus selected in ontology.

7.2.1 Transformation 'Argumental Structure' / 'Structure of Ontologies'

The passage of 'A*rgumental structure*' to ontology model requires rules organized into five groups:

-Rules to build entities classes, concepts classes, attributes of entities...

-Rules to define the relationships between objects from verbs.

-Rules to describe the role played by each verb, present in the structure, depending on the context in which it is located.

-Rules to describe axioms inherent to the various constraints that may be present in the structure.

-Rules for constructing axioms inherent constraints.

7.2.2 Calculation of the Closest Element

We consider a learning base containing examples of triplets taken as a set of tests.

What	Who	Whose	Where	When	How	Why	How_many	Segment	Ref
-------	-------	-------	------	-------	-------	---	------	S_1	L_1
-------	-------	-------	------	-------	-------	---	------	S_2	L_2
-------	-------	-------	------	-------	-------	---	------	S_3	L_3
.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.
-------	-------	-------	------	-------	-------	--	------	S_m	L_m

Where:

What , ***Who*** , …and ***How_many*** are the various variables that model the structure of triplets, *Segment* is the segment number containing the triplet and *Ref* is the link of triplet to its reference in ontology**.**

Table 2. Base of triplet examples with features

We consider input for our algorithm, a set of triplets expressed using arguments (What ...). We want to predict from this entry and a base of training examples (n examples), a reference to a structure of a triplet as arguments that are closest which can help to determine the corresponding ontological description in the ontology proposed.

This entry whose reference still unknown is then compared to all other structures triplets learned. We choose for the new data the majority class among its *K* nearest neighbors.

To find the *K* nearest triplets (expressed in '*argumental structure*') to classify, we have chosen the Levenshtein distance [18].

Algorithm 5. Calculation of the Closest Element

Input: Learning data **Argumenttrain** =($What^{train}$, Who^{train},$Whose^{train}$,$Where^{train}$,$When^{train}$,How^{train},Why^{train}, How_Many^{train}, $Segment^{train}$, $Reference^{train}$); *data whose the reference remains unknown and predict* ***Argumenttest*** = ($What^{test}$, Who^{test}, $Whose^{test}$, $Where^{test}$, $When^{test}$, How^{test}, Why^{test}, How_Many^{test}, $Segment^{test}$).

Output: element **Argumenttest(9).**

1: $ppvi \leftarrow \infty (i: = 1, n)$

2: **for** *i*:= 1 to *n* **do**

3: **begin**

4: **if** = $Argument_1^{train}(9) = Argument^{test}(9)$ **then**

5: **begin**

6: Calculate the distance Leveinsthein between $Argument_i^{train}(f)$ et $Argument^{test}(j)$

7: $d_j \leftarrow DIST_L(Argument_i^{train}(j), Argument^{test}(j))$

8: **end if**

9: **end for**

10: $ppv_i \leftarrow \Sigma \alpha_k . d_k \ (k = 1, 9)$

11: **end for**

12: $ppv\_retained \leftarrow Argmin_{i=1}^{n} ppv_i$ { ppv_retained contains the reference to the ontological description closest to the argument structure of the triplet to be inserted into the ontology considered}.

7.2.3 Integration of Segment (triplets) in Ontology

Integration of each segment in the ontology is processed by pipeline of various components as harmonization, transformation in symbols …, as shown in the figure below.

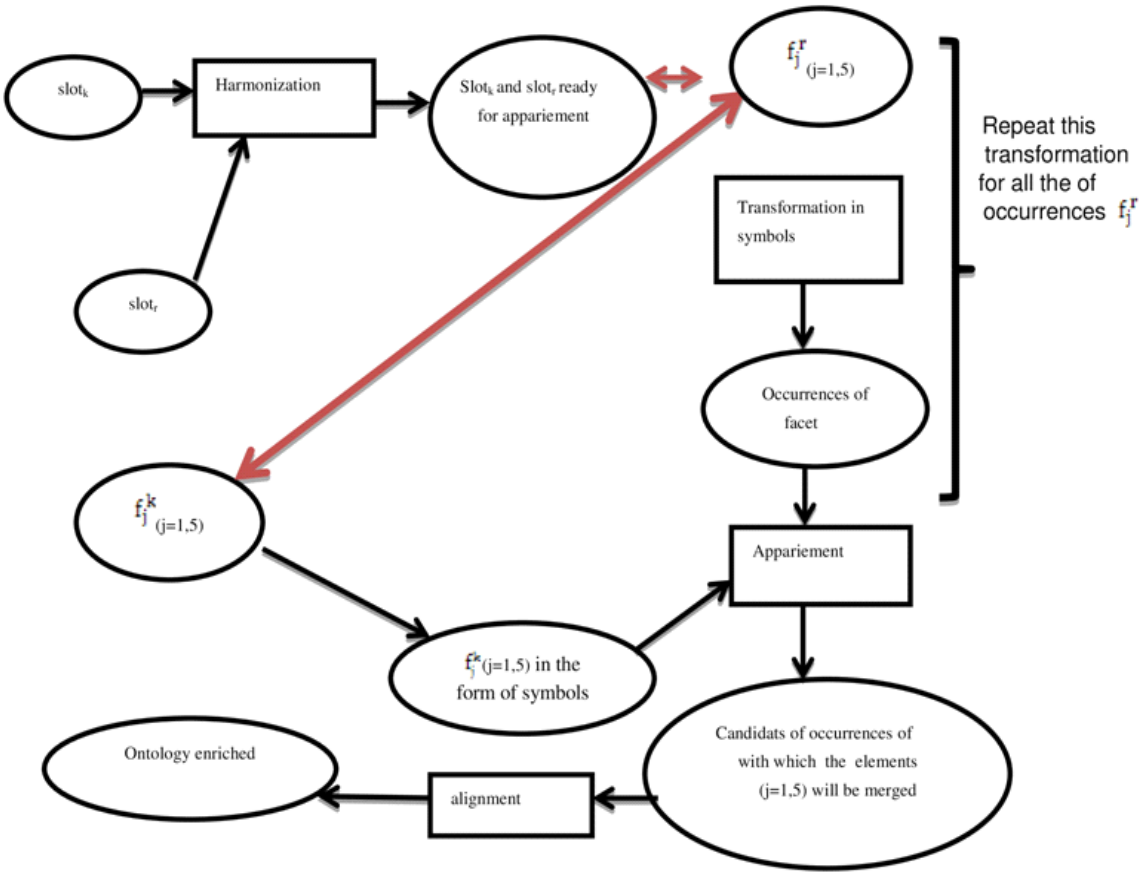


Figure 7. Integration of segments

We summarize, in the following, the different steps:

7.2.3.1 Harmonization

A facet f_i (i = 1, 5) is a sequence of terms and each term can consist of a token or more tokens. The task of harmonizing allows to normalize two specifications belonging to two facets of the same type and carry out the further treatments without ambiguities.

Example: We consider these two specifications

-If stock < threshold then…

-If stock quantity is less than one hundred units then …

After harmonization, we have

-If stock is below the threshold then …

-If stock quantity is less than 100 units then...

7.2.3.2 Transformation Symbols into Tokens

Let the facet f_i^k and let an occurrence of f_i^r to match with $f_i^k = \{t_1, t_2, t_3, \ldots t_n\}$ and $f_i^r = \{t_{n+1}, t_{n+2}, t_{n+3}, \ldots t_{n+m}\}$ where t_j (j = 1 ,…n + 1,…$n + m$) are the tokens forming the terms in facets.

In order to proceed the matching of facet f_i^k with occurrence f_i^r, we have to transform their elements in symbols.

Let $S = \{\alpha_1, \alpha_2, \alpha_3, \ldots \alpha_t\}$ a set of symbols and V the vocabulary (universe of discourse). We define d as the metric defined to VxV in [0, 1] , for this we have chosen the Hamming distance and δ a function of transformation defined to V in S whose we describe the transformation procedure of tokens in symbols :

Let T_r the set of tokens replaced by symbols, initially $T_r = \emptyset$.

We have $\delta(t_1) = \alpha_1$ and $T_r = T_r \cup \{t_1\}$

$\forall\, ti\ f_i^k \cup f_i^r$ and for each iteration i, we check:

- If $\exists\, t' \in T_r$ such $d(t_i, t') <$ threshold then $\delta(t_i) = \delta(t')$ else $\delta(t_i) = \alpha_w$ with $\alpha_w \in S$ and $T_r = T_r \cup \{t_i\}$.

To elucidate this principle of calculation, we consider this

illustration:

"*Study of the persistence of objects in a relational databases*" and "*To manage the schema objects in the database*" ,

we have:

f_i^k = {Study, of, the, persistence, of, objects, in, a, relational, databases} with 10 tokens

f_i^r = {To, manage, the, schema, objects, in, the, database} with 8 tokens.

The transformation of the two previous sequences in symbol sequences follows the following iterations:

Finally, the conversions of sequences f_i^k and f_i^r are: $\{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_2, \alpha_5, \alpha_6\ \alpha_3, \alpha_7, \alpha_8\}$ and $\{\alpha_9, \alpha_{10}, \alpha_3, \alpha_{11}, \alpha_6, \alpha_7, \alpha_3, \alpha_8\}$,these two sequences will be easily matched.

7.2.3.3 Matching Facet/ Triplet with Closest Occurrence /Facet

We have a first slot f_i^k(i =1, 5) that represents the triplet belonging to segment and a second slot f_i^r which is composed of a collection of occurrences. This slot is in the ontology and considered as the candidate closest to f_i^k for merge. At this level, all the facets of these two slots are transformed as sequences of symbols; it remains then to perform their matching. This matching problem can be modeled as the search for the longest common subsequence of two chains of terms.

Number of iteration	Token (t_i)	Corresponding symbol Value of δ	Comment
1st iteration	Study	α_1	
2nd iteration	of	α_2	
3rd iteration	the	α_3	
4th iteration	persistence	α_4	
5th iteration	of	α_2	$d(t_5, t_2) \leq$ threshold
6th iteration	objects	α_5	
7th iteration	in	α_6	
8th iteration	a	α_3	$d(t_7, t_3) \leq$ threshold
9th iteration	relational	α_7	
10th iteration	database	α_8	
11th iteration	To	α_9	
12th iteration	manage	α_{10}	
13th iteration	the	α_3	$d(t_{13}, t_3) \leq$ threshold
14th iteration	schema	α_{11}	
15th iteration	objects	α_6	$d(t_{15}, t_6) \leq$ threshold
16th iteration	In	α_7	$d(t_{16}, t_7) \leq$ threshold
17th iteration	the	α_3	$d(t_{17}, t_7) \leq$ threshold
18th iteration	database	α_8	$d(t_{18}, t_{10}) \leq$ threshold

Table 3. Transformation process of tokens in symbols

Algorithm 6. Matching Facet/ Triplet

Input: f_i^k ($i = 1, 5$) and (f_i^r)* ($i = 1, 5$) in the form of symbols.

Output: Determination an occurrence of (f_i^r)* can be integrated with f_i^k.

1: $j \leftarrow 0$

2: **for** i of 1 to 5 **do**

3: **for** m of 1 to N_i **do** {N_i , is the number of occurrences for each facet of f_i^r}

4: match (f_i^k , (f_i^r)^m, *size*) {(f_i^r)^m, is the occurrence having the value m as rank }

5: $T(i, m) \leftarrow$ *size* { size , length of the longest common subsequence}

6: **end for**

7: *rank* $\leftarrow$ *occurrence_rank* ($ArgMax_{m \in [1, nt]} T(i, m)$) {*rank*, the reference of the candidate occurrence to be merged with f_i^k}

8: Integrate (f_i^k , (f_i^r) *rank*)

9: **end for**

The procedure *match* compute the length of the longest common subsequence between f_i^r and f_i^k. This measure uses a recursive a recursive calculation:

$$\text{Length}(i,j) = \begin{cases} 0 \text{ if } i=0 \textbf{ or } j=0 \\ \text{Length } (i-1,j-1)+1 \text{ if } (i>0 \text{ and } j>0) \textbf{ and } (\text{element}_i(f_i^{r*}) = \text{element}_j(f_i^r)^m) \\ \text{Max } (\text{Length } (i, j-1), \text{Length } (i-1, j)) \text{ otherwise} \end{cases}$$

The cost of this algorithm is **O (a.b)** with a and b the number of elements (symbols) for f_i^r and f_i^k.

7.2.3.4 Integration Process

The principle of this process: we have, for this task, two facets, $f_i^k(i = 1, 5)$ (to integrate in ontology) and $f_i^r(i = 1, 5)$ (the occurrence candidate who will receive the previous facet). For each facet f_i^k and f_i^r $(i = 1, 5)$, if there is sufficient common elements then we align f_i^k else we insert f_i^k and this facet will be considered a new entry to the ontology.

For this, we summarize the correspondences studied in our approaches by the different situations:

Facet number 1	
Highlighting conflicts	**Ad hoc method**
Verbs sharing a close relationship + at least one common object	Terminology method with lexical and syntactic correspondences
Verbs sharing a close relationship + distinct objects	Terminology method with lexical and syntactic correspondences
Different verbs + objects close	Terminology method with lexical and syntactic correspondences
Inclusion of structures ex : v1(o1,o4) and v1(o1,o2,o3,o4,o5)	Method of comparing structures
Facet number 2	
Highlighting conflicts	**Ad hoc method**
Designation identical objects + attributes different	Semantic method + Terminology method with lexical and syntactic correspondences
Designation identical objects +common attributes	Semantic method + Terminology method with lexical and syntactic correspondences
Designation close objects + different attributes	Semantic method + Terminology method with lexical and syntactic correspondences
Inclusion of structures	Method of comparing structures
Facet number 3	
Highlighting conflicts	**Ad hoc method**
Named entities identical but the concepts used for their description are different	Comparison method of instances+ Terminology method with lexical and syntactic correspondences
Named entities different but the concepts used for their description are identical	Comparison method of instances+ Terminology method with lexical and syntactic correspondences
Identical named entities associated with structures having common elements	Comparison method of instances+ Terminology method with lexical and syntactic correspondences
Entity named in correspondence with a meta-entity named	Comparison method of instances
Facet number 4	
Highlighting conflicts	**Ad hoc method**
Close verbs + thematic roles different	Semantic method + Terminology method with lexical and syntactic correspondences
Different verbs + thematic roles identical	Semantic method + Terminology method with lexical and syntactic correspondences
Close verb+ whose some elements having thematic roles in common	Semantic method + Terminology method with lexical and syntactic correspondences
Facet number 5	
Highlighting conflicts	**Ad hoc method**
Specifications in correspondence described by concepts of different levels of representation	Comparison method of internal structures + Comparison method of external structures + semantic method
Specifications in correspondence described by common concepts	Comparison method of internal structures + Comparison method of external structures + semantic method
Specifications in correspondence with inconsistent assertions	Comparison method of internal structures + Comparison method of external structures + semantic method
Deductible assertions	Comparison method of internal structures + Comparison method of external structures

Table 4. Different situations of matching for each facet

For the integration conflicts between ontology elements, we can classify the resolution methods as follows:

Name of method	Corresponding function
Terminological method	It compares the labels of entities .It performs correspondence through the dissimilarity measures of chains while the lexical approach performs correspondence through the lexical relations (e.g., synonymy, hyponymy…).
Comparison method of internal structures	It compares the internal structures of concepts (e.g., interval value, cardinality of attributes…).
Comparison method of external structures	It compares the relationship of concepts with others. It is decomposed in comparison methods of concepts within their taxonomies and comparison methods of external structures taking account of cycles.
Comparison method of instances	It compares the extensions of concepts and compares the set of other concepts that are attached to it (occurrences …).
Semantic method	It compares the interpretations of concepts.

Table 5. Methods of conflict resolution

7.2.3.5 Transformation Operators for Integration Process

The identification of correspondences between the different structures and the definition of conflict between concepts and their semantic relationships must be validated by the domain expert. This will allow us to apply the operators of transformation between the elements: source / destination. And therefore use of the transformation operators for realizes the mapping task.

In [8], we have defined a set of operators for the integration problem of database schemas. These operators that we have adapted to the case of ontologies are intended to realize the integration and enrichment task of the ontology (Remove, add …), eliminate redundancy conceptual concepts ….

We divided these operators into four classes:

-Structuring Operators: This class is the basis of all defined operators, they include the growth and lowering functions which allow creating new elements or deleting others that already exist (links, attributes ...) .Their role is to manage the elements corresponding to the structure of the ontology.

Example: Creation / deletion of occurrences of facet, combining two instances of the same facet or more facets, moving an attribute, renaming an attribute...

-Hierarchization Operators: It is an extension of the operators working on the concepts defined in the ontology. They aim to generalize and specialize ontology concepts to build an inheritance hierarchy adapted to specific requirements of ontology and its users.

Example: Link two concepts for an inheritance relationship, merging of two concepts to build a new concept (generalization) , division of a concept to build two concepts (specialization),

-Populating Operators: Their role is to determine the extent of the objects by defining their qualification criteria or grouping objects or unbundling sets of objects.

Example: Adding / removing an occurrence of a facet according to a given criterion...

-Ensemblist Operators: They represent the set-traditional functions i.e the union (∪), the intersection (∩), the difference (-) ... Their role is to combine several source concepts to define new concepts.

Example: Gather several occurrences, remove some of occurrences...

We will find full details of these operators in [8].

8. Experimental Protocol

8.1 Types of Evaluation Considered

Our experimental model was designed to evaluate pipeline structure. We used the benchmark for test different parts constituting our approach. To evaluate the effectiveness of our proposed system, we first constructed a corpus containing articles. The accuracy was calculated using the ratio between the number of results answered by the gold standard and the total number of results answered by our system.

To rationally evaluate each module in the pipeline structure proposed in our approach, we consider four cases of evaluation:

- *Strong evaluation*: in this case, only the first step which is performed manually, the other three are performed by system note.
- *Mixed evaluation*: in this case the first two steps are carried out manually while the other two are realized by our system.
- *Low evaluation*: in this case, the first three steps are carried out manually while the latter is realized by our system.
- *Automatic evaluation*: in this case the four steps are carried out automatically by our system.

In addition to these evaluations, we add benchmark assessment experts who will serve us as a test for our approach.

8.2 Corpus Construction

Our text corpus contains a version containing various articles and topics ('Le Monde' with annual version '2013' in French). It has about 15,0000 sentences and the compound words are a total of more than 9% of lexical units in the corpus. All articles were processed to separate raw text content (containing only paragraph boundary information) from formatting and other page elements. Meta information, such as page title and title variants (obtained by processing redirection page links), category labels, hyperlinks within the text etc. were retained in separate files to facilitate later processing. We collected newspaper articles to have a corpus of a certain size. Then the software 'Open Source' Unitex [21] was used to conduct research and build concordance files to isolate sentences containing verbs to perform cutting phrases in two parts: left syntagm, right syntagm [7].

8.3 Evaluation

To measure the performance of our approach, we create two bases, each containing the same set of sentences extracted from corpus.

-*The first base* is used by the experts for the construction of ontology, it is considered as a benchmark for the evaluation of our approach. The domain experts are the curators of the corresponding Gold Standard to assess the domain coverage corresponding to the corpus. We choose a number of this base intended for the evaluation of our approach. A group of experts (three in number) treat manually this base for building ontology and case there would be a relative consensus between experts, we select their solution as gold standard.

-*The second base* is considered a base of test, it is used as an input for our approach for its evaluation. In what follows, we considered two cases of evaluation:

We asked the experts to build the ontology of two ways:

- In following the four steps of our methodology manually.

- In applying their knowledge using an empirical method.

For each of these two cases mentioned, we will evaluate our automated approach to their results.

The measure recall, precision and f-measure are used for comparing a reference building ontology (Gold Standard) with our approach of building ontology (our system). Precision and recall are defined as follows:

Let *N*1 be the set of all objects relevant of our ontology building system and *N*2 the set of concepts of the gold standard ontology. The lexical overlap is equal to the ratio of the number of concepts shared by both ontologies i.e. the intersection of these 2 sets.

P (Our approach, Gold Standard) = (lexical overlap)/(lexical overlap +card(N2-N1)) =card(N1∩N2)/card(N1∩N2)+card(N2-N1)) noting in passing that the sign '-' denotes the ensemblist difference.

R (Our approach, Gold Standard) = (lexical overlap)/(lexical overlap +card(N1-N2))

The F-measure is used for giving a summarizing overview and for balancing the precision and recall values. The Fmeasure is the harmonic mean of P and R.

F (Our approach, Gold Standard) = 2 • P(Our approach, Gold Standard) • R(Our approach, Gold Standard)/(P(Our approach, Gold Standard) + R(Our approach, Gold Standard))**.**

8.3.1 Evaluation Using Benchmark Ontology Issued by Our Approach

The construction of the ontology that will be taken as reference, in this case, for the evaluation of our approach willbe obtained through the application of our approach. The experts will follow the sequence of stages that will lead to the construction of ontology Gold Standard.

-Strong Evaluation:

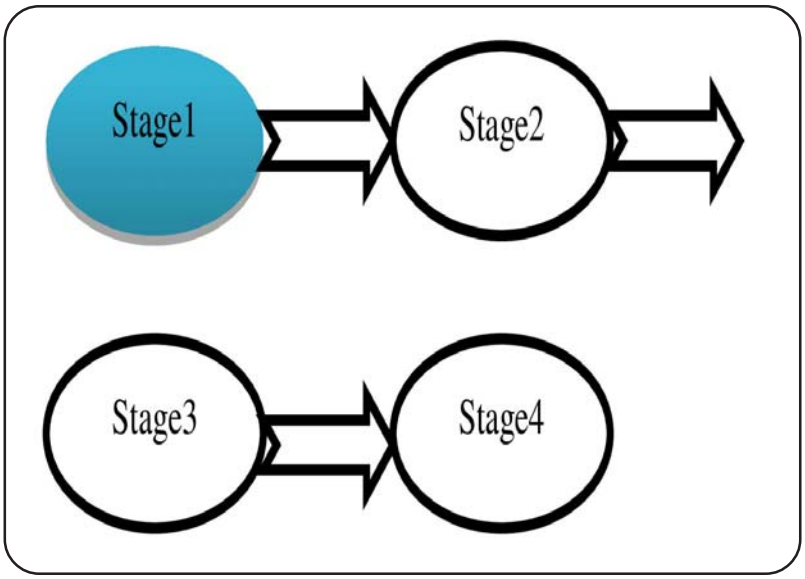


Figure 8. Strong evaluation

In this type of evaluation, we assume that the first phase is already done by the experts, remains to evaluate our system based on the three other stages. For this, we take as input to our approach, the results provided by the experts for the first stage. The evaluation of our approach for this case gives the following results:

	Precision	Recall	F-measure
Facet1	79,42	78,87	79,14
Facet2	76,70	75,15	75,91
Facet3	78,54	76,40	77,45
Facet4	77,19	75,09	76,12
Facet5	70,68	68,28	69,45

Table. 6 Recall, precision and F-measure for building ontology (Stage2+ Stage3+ Stage4)

Mixed Evaluation:

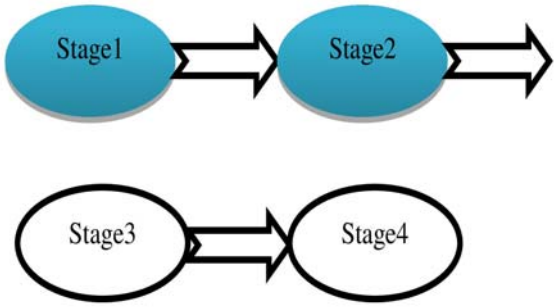


Figure 9. Mixed evaluation

This type of evaluation takes as input the results provided by the experts for the first two stages. The evaluation of our approach in this case gives the table:

	Precision	**Recall**	**F-measure**
Facet1	82,28	81,05	82,63
Facet2	78,14	77,78	77,95
Facet3	81,93	79,95	80,92
Facet4	80,62	78,23	79,40
Facet5	74,47	72,36	74,41

Table 7. Recall, precision and F-measure for building ontology (Stage3+ Stage4)

Low Evaluation

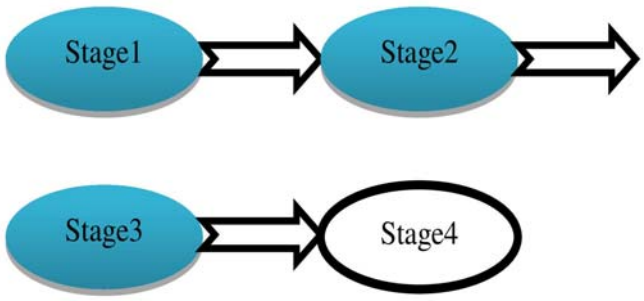


Figure 10. Low evaluation

This case evaluation is restricted to the last stage, the first three spots were performed manually by experts, where from:

	Precision	Recall	F-measure
Facet1	95,42	93,69	94,54
Facet2	91,07	90,81	91,56
Facet3	94,65	93,38	94,01
Facet4	93,16	92,57	92,86
Facet5	86,25	84,11	85,04

Table 8. Recall, precision and F-measure for building ontology (Stage4)

- Automatic Evaluation

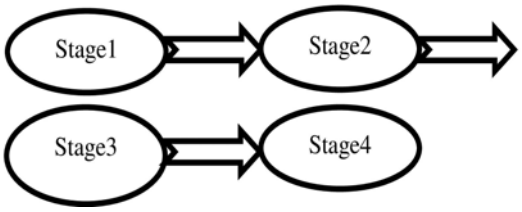


Figure 11. Automatic evaluation

This type of assessment allows to fully test our approach for building ontology. All stage is taken entirely by our approach and the final result generated by pipeline stages is compared with the results of the Gold Standard:

	Precision	Recall	Fmeasure
Facet1	78,81	76,10	77,43
Facet2	75,70	72,93	74,28
Facet3	76,07	74,28	75,16
Facet4	75,16	73,87	74,50
Facet5	68,94	66,34	67,61

Table 9. Recall, precision and F-measure for building ontology (Stage1+Stage2+Stage3+Stage4)

8.3.2 Evaluation Using Benchmark Ontology Issued by the Know-how of Experts

The construction of the ontology to be used as a reference for the evaluation of our approach will be obtained empirically that is to say, by the know-how and experience of experts .It will be the second ontology Gold standard. The evaluation results in this case are summarized in the following table:

	Precision	Recall	F-measure
Facet1	74,89	72,43	73,70
Facet2	71,75	70,81	71,27
Facet3	73,51	72,18	72,83
Facet4	72,64	70,33	71,87
Facet5	66,08	63,97	64,78

Table 10. Recall, precision and F-measure for building ontology with comparison to Benchmark ontology issued by the know-how of experts

8.4 Results and Discussion

The results found in the various tables are obtained by comparing the ontology generated by our system to the ontology built by the experts (Gold Standard). We repeated this evaluation process on other texts randomly selected from the same corpus and the results are always close to the results of these tables. According to [19] Comparing two ontologies can be done at two levels: lexical and conceptual. Lexical comparison assesses the similarity between the set of terms denoting concepts of the two ontologies. At the conceptual level, the taxonomic structures and the typology of relations are taken into consideration for comparison of two ontologies. To simplify the comparison task of ontology issued from our approach with that obtained by the experts, we opted for the lexical coverage method.

In all tables, we note that the results obtained for the facet 5 are underperforming compared to other facets. The reason they gave is undoubtedly related to the complexity of this facet. However, the results for the facet 1 and the facet 3 are high due to decoupage of phrases in triplets. This type of phrase decoupage used to extract the verb and locate the associated nominal syntagms.

The results of table 6, 7 and 9 are approximately identical this is explained by the fact that every stage of our approach except the last stage have significant and encouraging results.

The same applies to the table 8, which confirms the performance of process of ontology building (step 4). The calculation of F-measure in Table 9 shows that there is a correlation between the manual task performed by experts using our approach (the four steps of our approach) and the results generated by our system. The average value of Fmeasure is equal approximately 74%. Such a result is very acceptable especially for a combinatorial problem like ours. Finally the results provided by our approach differs slightly from empirical process based on the know-how of experts (see table 10), this shows that results of our approach remains valid whatever the work conducted by the experts for the process of ontology building from texts.

In sum, we could conclude that the source of errors came from the ambiguous instances related to dissimilar concepts, name conflict and conflict of granularity concepts.

9. Comparison of Our Approach With Current State of the Art

In order to validate the effectiveness of our approach, we compared the ability of our approach to building extended ontologies with the set of methods presented in the related work. For this, we draw up a table with the different approaches using more objective and significant criteria (7 in number) that will serve us as a strong clue of comparison for assess our work compared to current work.

According to the comparative study of the table above, we can say that our approach ensures a acceptable accuracy with a very high level of abstraction and genericity. What gives promising results on all criteria in comparison to other related approaches (low genericity, moderate automation ...).

10. General Conclusion

In this work, we presented a novel framework for the construction of terminological ontologies through a text corpus. The framework proposed in this study is based on the pipeline using deep linguistic information. It includes a new parsing strategy of a topic segmentation of texts and introduces several heuristics for pattern discovery for automatic extraction of key-concepts. The patterns to be learned are for extracting key-concept where each key-concept has an associated relevance value, which represents how relevant the key-concept in the text. To do this, the framework exploits additional linguistic resources to obtain a more accurate matching. Based on this matching, several metrics are combined to obtain some objective measures. These key-concepts are introduced by integrating information heuristics in ontology. The process has been evaluated through a case study conducted in the domain of News paper (Le Monde in French), showing good results, and demonstrating that the use of techniques of natural language processing represents a promising approach for building and enrichment of ontologies.

Our perspective, then, is to use semantic web mining techniques and to restructure web pages in order to implement an adaptive web based on the semantic structure, content and services. Such a process greatly simplifies the problem for ontology building from web pages. We can conclude that we must take into account various learning sources like on-line linguistic resources and structure regularities in web sites to go further in the implementation.

Approach	Tools used	degree of automation	Domain of experimentation	Accuracy	Pros	Cons
Document Ontology Extractor	Latent Semantic Indexing based on SVD (Singular Value Decomposition)	Automatic	Corpus containing a collection of text documents	70 %	The system takes text documents and generates useful ontology and it is possible to view and modify this ontological Structure with graphical user interface	The work will be more efficient if the author has added some heuristics for develop the relations between the concepts
Variabilité des performances des outils de TAL et genre textuel. Cas des patrons lexico-syntaxiques (**in french**)	Base of lexical relation patterns+ CAMÉLÉON+NPL	Semi automatic	Eight different corpora (scientific articles, articles extracted from the Encyclopedia Universalis…)	The efficiency varies strongly depending on the test corpus varying 10% to 58%	Transition of generic base of patterns towards a reusable base	how to determine the validity of a pattern ?
Discovering Relations among Named Entities form Large Corpora	Unsupervised method for relation discovery by clustering pairs of named entities	Semi automatic	Corpus compound of one year of The New York Times (1995)	Between 75 % and 80 %	Method proposed does not need the richly annotated corpora required for supervised learning	The method by tuning parameters used is inadequate
Extending Ontology Tree Using NLP Technique	Natural Language Processing (NLP) and a tool of Information Retrieval (IR)	Good automatic topic identification system	200 web documents selected randomly	95.5% on a limited test achieved on a set of 200 documents	The idea to incorporate an external linguistics knowledge-base (WordNet) to enrich the ontology concepts is a good investigation for the discussed issue	Not all words which are semantically related to the words concepts considered are suitable to be used as the extended ontology
Semi-Automatic Ontology Extension Using Spreading Activation. Journal of Universal Knowledge Management	Spreading activation model	Semi-automatic	Corpus gathered from a large sample of news media sites specialized	No cited	Good tool for refine ontologies by mining textual data from the Web sites	The using of spreading is achieved on a seed ontology on "climate change" remains insufficient for generate ontologies in general case
Sofie : a self-organizing framework for information extraction	Maximum satisfiability problem (MAX SAT MODEL)	Automatic	Semi-structured sources from Wikipedia and with unstructured free-text sources from the Web	Between 90 % and 95 %	Good results for the structured Internet documents case.	Good reconciliation combining a pattern-based information extraction, entity disambiguation, and ontological consistency constraints into a unified framework
Our approach	A pipeline model comprising a panoply of complex heuristics based on computational linguistics and statistics	We define two versions , one automatic and the other semi automatic with a precision calculation	100 articles randomly drawn from text corpus ('Le Monde' with annual version '2013')	74%	An original parsing strategy of a topic segmentation of texts+ definition of two novels models , one for assigning semantic roles for sentence the other for the for the representation of ontologies	in the case where the system is fully automatic, it is then necessary to improve the accuracy

Table 11. Analysis of the performance of different approaches related with our approach

References

[1] Faatz, A., Steinmetz, R. (2002). Ontology enrichment with texts from the www, *In*: Semantic Web Mining 2nd Workshop at ECML/PKDD, Helsinki, Finland.

[2] Aussenac-Gilles, S., Despres, S., Szulman. (2008). The terminae method and platform for ontology engineering from texts, Proceedings of the conference on Ontology Learning and Population: Bridging the Gap between Text and Knowledge.

[3] Aussenac-Gilles, N., Jacques, M.-P. (2006). Variabilité des performances des outils de TAL et genre textuel. Cas des patrons lexico-syntaxiques. Dans : Traitement Automatique des Langues, Numéro spécial Non Thématique, Paris : Hermès Sciences, 47, (2).

[4] Auger, A., Barrière, C. (2008). Pattern based approaches to semantic relation extraction: *a state-of-the-art Terminology*, 14 (1) p. 1–19.

[5] Alonso, A., Millon, C., Williams, G. (2011). Collocational networks and their application to an E-Advanced Learner's Dictionary of Verbs in Science (DicSci). *In:* Proceedings of eLex, p. 12 - 22.

[6] Benafia, S. (2012). Apport du langage naturel dans l'indexation des images Mémoire de master -Université de Biskra.

[7] Benafia, A., Maamri, R., Sahnoun, Z. (2013). An Indexing Approach based on a Hybrid Model of Terminology-extraction using a Filtering by Elimination Terms *Journal of Advances in Information Technology*, 4 (1) 28-39.

[8] Benafia, A. (1995). Une nouvelle approche de modélisation des systèmes d'information, Thèse de magistère Université de constantine.

[9] Berroyer, J.-F. (2004). TagEN, un analyseur d"entités nommées : conception, développement et évaluation . Mémoire de D.E.A. d'Intelligence Artificielle, Université Paris-Nord, 2004.

[10] Brown, G., Yule, G. Discourse analysis. Cambridge: Cambridge University Press.

[11] Chakravarthi, S. (2001). Document Ontology Extractor, Applied research in Computer science, Fall.

[12] Dong, W., Charikar, M., K. Li. (2008). Asymmetric distance estimation with sketches for similarity search in highdimensional spaces. SIGIR.

[13] Fabre, C., Bourigault, D. (2001). Linguistic clues for corpus-based acquisition of lexical dependencies. 2001 Conference, UCREL Technical Papers, 13, Lancaster University, p. 176-184 , 2001.

[14] Fillmore, C. (1982). Frame semantics, In the linguistic Society of Korea (Ed.), Linguistic in the morning calm 111-137 Seoul:Hanshin Publishing Co.

[15] Hasegawa, T., Sekine, S., Grishman, R. (2004). Discovering Relations among Named Entities from Large Corpora. *In:* Proceedings of ACL.

[16] Hanks, P. (2010). Terminology, Phraseology, and Lexicography. In A. Dykstra and T.Schoonheim (eds.), *In*: Proceedings of the XIV Euralex International Congress, 6 - 10 July, Leeuwarden, x.Ljouwert: Fryske Akademy / Afuk.

[17] Liu, W., Weichselbraun, A., Scharl, A., Chang, E. (2005). Semi-Automatic Ontology Extension Using Spreading Activation. *Journal of Universal Knowledge Management* (1) 50–58.

[18] Levenshtein, V. I. (1965). Binary codes capable of correcting deletions, insertions, and reversals, Doklady Akademii Nauk SSSR, 163 (4) 845-848, (Russian). English translation in Soviet Physics Doklady, 10 (8) 707-710, 196.

[19] Maedche, A., Staab, S. (2002). Measuring similarity between ontologies, *In:* Proceedings of European Knowledge 24 Acquisition workshop (EKAW), Springer.

[20] Paumier, S. (2003). De la reconnaissance de formes linguistiques à l'analyse syntaxique, Thèse de +doctorat, Université de Marne-la-Vallée.

[21] Palmer, D. (2010). Text Pre-processing, Handbook of Natural Language Processing, Second Edition, CRC Press, Taylor and Francis.

[22] Romary, L., de la Clergerie, E. (2004). International standard for a linguistic annotation framework. *Natural Language Engineering*, 10 (3/4) 211–225.

[23] Suchanek, F. M., Sozio, M., Weikum, G. (2009). Sofie: a self-organizing framework for information extraction. *In*: Proceedings of the 18th International Conference on World wide web, p. 631–640. ACM.

[24] Sagot, B., Fiser, D. (2008). Construction d'un WordNet libre du Francais à partir de ressources multilingues, *In*:TALN, Toulon.

[25] Tatane, K., Er-raha, B., Mouhim, S., Cherkaoui, C. (2013). Semi-Automatic Enrichment Approach of 'Domain Ontology' by using TALN Tools, *International Journal of Innovative Research in Computer and Communication Engineering* (An ISO 3297: 2007 Certified Organization) 1 (10).

[26] Tollari, T. S., Rosni, A., Enyakong, T. (2001). Extending Ontology Tree Using NLP Technique. *In*: Proceedings of National Conference on Research & Development in Computer Science REDECS.

[27] Tyler, S. (1983). The said and the unsaid. mind, meaning, and culture. New York, San Francisco, London: Academic Press.

[28] Uschold, M., Gruninger, M. (1996). Ontologies: principles, methods and applications. *Knowledge Engineering Review*, 11 (2) 93-155.

[29] Welty, C., Murdock, J.W. (2006). Towards Knowledge Acquisition from Information Extraction Book , The semantic Web :LISWC 2006 , LNCS 4273. p. 709-722, Springer-Verlag Berlin Heidelberg.

[30] Zhong, P., Chen, J. (2008). Web Information Extraction Using Web-specific Features , *JDIM* 6 (3) 235-243.