Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 73 (2015) 16 – 23

Procedia Computer Science

The International Conference on Advanced Wireless, Information, and Communication Technologies (AWICT 2015)

Dynamic Evaluation of Ontologies

Kheira Lakel[a], Fatima Bendella[b]

[a]*Departement of informatic, Universit des sciences et de la technologie d'Oran - Mohamed Boudiaf BP 1505 Oran El M'Naouer - ORAN,ORAN 31000, Algeria*

[b]*Departement of informatic,Universit des sciences et de la technologie d'Oran - Mohamed Boudiaf BP 1505 Oran El M'Naouer - ORAN, 31000, Algeria*

Abstract

The automatic construction of ontologies from texts is a topic of continued and open research, their construction requires both a study of human knowledge, methodologies and tools to retrieve the text content. As the content of these resources is dynamic, they can be thought of as finished products and refined, which remain stable when completed. The field of ontology construction needs to go towards more dynamic, more view of ontologies is to increase intelligence in many applications such as information retrieval, semantic indexing and semantic annotation. Ontologies are software modules whose development is based on the same principles as those applied in software engineering. There are several approaches for evaluating ontologies, some are based on learning methods from the corpus, using the networks head-Expansion or other semantic networks for identify concepts and relationships. But the automation of ontology construction, actually, is a scientific lock for many applications. In this paper, we propose an approach that combines these tools to improve the process of automatic co-construction of ontologies from a corporus. DEO (Dynamic Evaluation of Ontologies) is a system dedicated to ontology construction from texts using a cooperative learning based on a multi agents structure.It uses the mechanism of extraction of relations Dynamo and more it uses the terminology WordNet1.2 to identify concepts, relationships and a storage module to save the changes of the ontologist in order to be reused.

Peer-review under responsibility of organizing committee of the International Conference on Advanced Wireless, Information, and Communication Technologies (AWICT 2015)

Keywords: Construction of ontology, ontology learning, ontology engineering, Semantic Web, multi-agent systems, ontology,

1. Introduction

Ontologies appeared as a key for automatic handling of information at the semantic level which makes them a central component in many applications, and are called to play a crucial role in the Semantic Web which is the next evolution of the Web. It is a set of technologies designed to make the content resources of the World Wide Web accessible and useable by programs and software agents, which should facilitate access to information for users. One of the major success of the Semantic Web is the availability of ontologies which are representations of formalized

E-mail address: kheira.lakel@univ-usto.dz

Peer-review under responsibility of organizing committee of the International Conference on Advanced Wireless, Information, and Communication Technologies (AWICT 2015)

doi:10.1016/j.procs.2015.12.043

knowledge exploitable by computer systems for their communication. Unfortunately their construction is generally tedious and expensive, and their maintenance poses problems until now underestimated[1]. Ontologies are fundamental building blocks of the Semantic Web and frequently used as components of advanced information systems, for all these reasons, it is essential to build them and keep them updated. To achieve this goal, the field of ontology construction needs to go towards more dynamic. In recent years, several engineering platforms for learning ontologies[2] have been developed incorporating some of the tools and methodological elements.In this paper, we present DEO, it is a system of building ontologies based on learning, re-engineering and cooperative methods. Hybridization between these three methods allowed us to make this construction dynamic. To distribute the control and knowledge in the ontology we integrated multi-agent systems (MAS), using MAS is justified by their ability of adaptation to changes and developments[3]. The MAS due to their distributed nature, the reasoning is done locally, the addition and removal of agents when operating, considerably facilitate the system's adaptability to any changes.

2. Methods for ontology construction

The ontological engineering is a subdomain of knowledge engineering that studies the process of ontology development, methods and methodologies for the construction[4]. Ontological engineering[5] is a research methodology which gives us the design rationale of a knowledge base, kernel conceptualization of the world of interest, semantic constraints of concepts together with sophisticated theories and technologies enabling accumulation of knowledge which is dispensable for knowledge processing in the real world . Thus, an ontology defines the terms and the relations of the basic vocabulary of a domain and the rules that show how combine terms and relations so as to extend the vocabulary[6]. In the literature[7][8], Many methods are proposed for building ontologies.

Methods for ontology building from scratch, they are based on the extraction of common knowledge manually into the different sources, then they use techniques of Natural Language Processing (NLP) and acquisition of knowledge to generate new knowledge from those acquired in the previous step. *Methods for cooperative construction of ontologies*, Ontology must be a consensus and be accepted by its user community. These methods therefore adopt a collaborative approach for construction including the intervention of persons located in different places. *Methods for re-engineering of ontologies*, Re-engineering of ontologies is the process of rebuilding ontologies and linking a conceptual model of an ontology already implemented in another being implemented. *Methods of learning ontologies*, they consist in improving the construction of ontological components by introducing plug-ins in the process of ontology development, these plug-ins can be text and knowledge bases The proposal of a methodology for building ontologies is based on learning, re-engineering of ontology and cooperative approaches that is, hybridization between these last three methods. The concept of MAS has been integrated to make their buildings dynamic and to distribute the control and the knowledge. This is justified by the following reasons, that the automatic construction of ontologies by learning from textual corpora is generally based on the text itself, and the produced ontology is rich in terminology but it is not a finished product because it limited to written content (it doesn't contain all the knowledge manipulated in a field). To conceive ontologies semantically richest, it was proposed to extend the ontology learning methods in the construction of ontology by taking into consideration the content of texts to build an initial kernel of ontology and for enriching the ontology obtained from methods of re-engineering of ontology were used to add the implicit knowledge by exploiting external resources. But they can not guarantee that all knowledge of a domain are added in this step. In this case the intervention of an expert is used to enrich, to correct and improve the structure of knowledge. Generally, the process of building ontologies is complex, involving multiple stakeholders in different phases. In recent years, MAS have become important in the field of computer science. The recent strength of this paradigm comes from flexibility and variety of interactions. MAS technology offers the possibility to specialized agents to execute in parallel and concurrent manner[3]. Thus, agents use their learning capacity to adapt and interact with others, and as the process of building ontologies is complex, distributed and requires learning to maintain these ontologies, we can use MAS for their constructions.

3. Related works

Currently, in the continuity of knowledge engineering approaches; the works focuses on the integration of language processing tools and NLP methods to build platforms as Text2Onto[9]. Another trend, related to the pressure of the

Semantic Web requires rapid availability of ontologies, to populate and to introduce more automation by learning techniques.

ASIUM[10]stands for Acquisition of SemantIc knowledge Using Machine learning Method allows the acquisition of semantic knowledge from the corpus by unsupervised learning where it uses syntactic relations to determine the relationship verb-complement from syntactic regularities and then to interactive validation phase. Its objective is to group the terms for create pertinent semantic classes,the final taxonomy must be validated by an expert to formalize ontologies. This approach uses a bottom-up method in width to form the concepts of the ontology level by level. If a class is badly built, we must find the stage of the reasoning that led to this erroneous result and manually edit the corresponding class. But in this case, all the consecutive steps in the creation of the modified class will be lost and must be recalculated taking into account the change.

Dynamo[1] stands for DYNAMic Ontologies is a tool that allows the co-construction of ontologies. It is organized into several modules. The processing module of textual corpus supports the preparation of inputs MAS. It uses Syntex tool an extractor of terms which is selected primarily for its robustness and the large quantity of information extracted from both proposals for syntactic dependencies and distributional analysis. We focused on the network head-Expansion created by this tool, which is an interesting structure for a classification system. DYNAMO MAS is formed by two types of agents: a TermAgent reflects the terminology component of the ontology and a ConceptAgent represents the conceptual component of the ontology. Each TermAgent manages the lexical relations of which it is source or target. Also, each ConceptAgent manages the conceptual relationships that it is source or target. In the output, Dynamo creates an ontology as an owl file. Despite the modifications attached to the inputs and to co-construction process, produced ontology is not rich in terminology and conceptual plans. Also we noticed that there is not a process that disambiguated the sense of ambiguous term this mean that, the ontologist must be based on his own knowledge to select the relevant sense. As the objective of the disambiguation of the term (concept) is to improve the relevance of candidate terms. Moreover improvements that are made by ontographe will be lost (no a storage module) so for a corpus like the ontographe must repeat the same changes. OntoLearn stands for Ontology Learning is a method for the dynamic evaluation of ontologies from text through learning. It is based on statistical methods to identify candidate terms, then it uses generic ontological resources such as Wordnet for aggregation of these terms in order to obtain a domain ontology[11].

4. Proposed approach : Dynamic Evaluation of Ontologies (DEO)

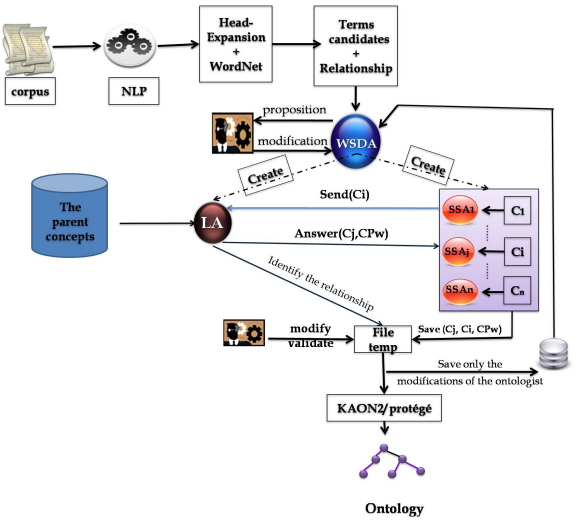

Fig. 1. DEO architecture

We propose an approach based on cooperative learning (system-users) for building ontologies dynamically **DEO** (**D**ynamic **E**valuation of **O**ntologies). This system is based on a set of agents that cooperate and interact to evaluate

the ontology see Figure 1. The DEO System integrates four main processes:

1) Extraction process of candidate terms

We done an analysis of terminology using an NLP tool to identify indices of language knowledge (terms, lexical relations, semantic classes, etc.). Like NLP tool, we use TreeTager[12] to tag texts. Next, we used WN 1.2(Wordnet 1.2)[13] terminology to identify concepts and semantic relationships. Moreover we used the network Head-Expansion[1] to determine the terms and lexical and / or semantics relations that are not defined in this terminology, as this process a knowledge base created dynamically from file.temp . This database should contain only new knowledge or modified knowledge by ontologists whose aim is to preserve the information and reused them in a similar situation. Moreover, the ontologist may intervene in this phase to validate, modify, delete or add concepts and semantic relations.

2) Process of identifying concepts

The concept is necessarily unique. In this phase WSDA agent (Agent Word Sense Disambiguation) is created to disambiguate the sense of an ambiguous term. Ambiguity is the property of a word or a sequence of words to have several senses or more grammatical analyses possible. This is also the character of a difficult situation to understand. We talk about semantic disambiguation when each word is linked in a given context to single given definition. This objective is one of the objectives of the consistency of natural language[14].To remove some ambiguity, the agent WSDA calculates the semantic distance between all possible meanings for a term i and all possible meanings for another term j such that i! = j, and it chooses the most similar meaning. It selects the nearest node common to both terms, then it found to each sense its synonyms and definitions from WN. At the end of this phase, WSDA created for each synset an SSA agent (Agent SynSet) and for common nodes it create a LA (Link Agent). Moreover, the ontologist can intervene to add, modify or validate the synsets and common nodes.

The semantic disambiguation step

Pretreatment: In this step, we created two sets. The set M_{wn} whose WSDA which inscribed all the words known by WN1.2 and The set M whose WSDA which inscribed all unknown words by WN1.2.

For known words by WN1.2 we used the Wu-Palmer measure[15] because in a domain of concepts, the similarity is defined with respect to the distance between two concepts in the hierarchy and also by their position relative to the root. The similarity between C_i and C_j is:

$$ConSim(C_i, C_j) = \frac{2 * N3}{N1 + N2 + 2 * N3} \tag{1}$$

Either N_1 and N_2 correspond to the number of is_a relationships between C_i and C and the number of is_a relationships between C_j and C and N_3 is the number of arcs is_a since the common ancestor C to the root of the taxonomy (in WN1.2, the root is 'entity').This measure has the advantage of being simple to implement and to have such good performances than other similarity measures[16].The similarity values obtained are normalized in an interval between 0 and 1, with 0 corresponding to a maximum dissimilarity, and 1 representing a complete identity. The similarity $ConSim(C_i, C_j)$ obtained is high when the keywords are close in the tree of the external resource (WN1.2).

Rule 1: Similarity estimation

Witch we have effected several tests for choose the minimum threshold which allow to identify the maximum of the relevant terms and n is the number of words in M_{wn}

If $\Sigma_{0<=j\wedge j\neq i}^{n-1} \frac{ConSim(C_i, C_j)}{n-1} \leqq threshold$ then remove the word C_i (because the word C_i is dissimilar to other words).

Rule 2: Identification of synsets and refinement

In this step, the agent WSDA consists to search from candidate words the synonyms for building the synsets. Either M_{wn}, S, SYN three sets: M_{wn} is the set of candidate terms extracted from the corpus, S is the set of candidate terms enriched by WN1.2; such as initial S is empty and SYN is the synsets moreover We defined the function $fdepth(C_i, C_j)$, which $C_i \epsilon SYN_i$ and $C_j \epsilon SYN_j$, this function allows to calculate the distance between C_i and C_j in number of arcs.

Example

Either the two words: *document* and *papers*

The word *document* has four senses:

1. document, written document, papers - (writing that provides information (especially information of an official nature))
2. document - (anything serving as a representation of a person's thinking by means of symbolic marks)

3. document - (a written account of ownership or obligation)
4. text file, document - ((computer science) a computer file that contains text (and pos sibly formatting instructions) using seven-bit ASCII characters)

Then the possible synsets are:
SYN1={document, written document, papers}, SYN2= {document}, SYN3= {document} and SYN4={text file, document}
The word *papers* has one sense only:

1. Document, written document, papers - (writing that provides information (especially information of an official nature))

Then the possible synsets are: SYN1= {document, written document, papers}
fdepth (SYN1,SYN1) = 0 fdepth (SYN2,SYN1) = 13
fdepth (SYN3,SYN1) = 4 fdepth (SYN4,SYN1) = 9
If $fdepth(C_i, C_j) = 0$ then C_i and C_j are synonyms So $C_i, C_j \epsilon SYN$ (SYN = SYN1 = SYN2) then $S = S + SYN1$ and $M_{wn} = M_{wn} - X_i + R$ where R is the smallest generalization which subsumes X_i and X_j ; In our example the smallest generalization is the synset: Writing, written_material, piece_of_writing - (the work of a writer; anything expressed in letters of the alphabet (especially when considered from the point of view of style and effect); "the writing in her novels is excellent"; "that editorial was a fine piece of writing"); So $R = Writing$; We repeated this step for all words $X_i, X_j \epsilon M_{wn}$ until the condition $fdepth(C_i, C_j) = 0$ become false and If $fdepth(C_i, C_j) \neq 0$ then WSDA chooses the SYN which what $fdepth(C_i, C_j)$ is minimal and $S = S + SYN1 + SYN2 + R$; where R is the smallest generalization which subsumes X_i and X_j.
For unknown words by WN1.2 the agent WSDA adds a word X to S in the following cases:
Rule 3: If X is a compound word, WSDA used the link Head / Expansion to identify the Head (X). So if $Head(X) \epsilon S$ **then** $S = S + X$
Rule 4: If X **is in relation with another term in** S**,** $X ® S$ **where the ® is a relationship of subsumption or acronymie type.**
The ontologist can intervene to validate the synsets and the common nodes. In the end of this phase, WSDA created for each synset an SSA and for the common nodes it created a LA.
3) Hierarchization process of concepts
The agent SSA send the message (SSA_i), by this he asked the agent LA to send their brothers and their father. Then, the agent LA responds by message Answer (SSA_i, C_j, C_i, CPw) where C_j is the brother of C_i and CPw is the nearest common father in the hierarchy of WordNet for C_i and C_j. Next, the agent SSA saves tuples (C_j, C_i, CPw) in a file .Temp and agent LA specifies the semantic relationship between these concepts and so on for other concepts. Similarly, ontologists may be involved in this step to improve and remove tuples and / or relationships. In this phase, a storage module saves only like mentioned earlier the changes made by the ontologists able to be reuse.In the end of this phase, we associate to the concept entity the agent TOP (an agent who has no parent) where all pivotal concepts of these hierarchies constructed are directed to the concept of entity of the agent TOP.
4) Process of creation of the formal ontology
In this phase, the ontology must be saved in a formal format (OWL file, XML, RDFS), in using platforms KAON2 [17] or editors like protg [18].

5. Evaluation and Analysis

To evaluate the properties of our DEO system, we have chosen, in the context of experimentation, datasets of different sizes in the field of security-cryptography from the corpus 20 news groups [19]. The collection contains 500 words, after executing the first process, we obtained a database of 184 candidate terms of words. This database is divided into two groups: The known terms by WN1.2 as: key, system, encrypt, decrypt, etc. and The unknown terms by WN1.2 like: crypto system, RSA, ect.In this phase, semantic relations are extracted from the lexical analysis of a sentence such as synonymy, hyperonymy meronymy,... etc. To enrich the ontology by new relationships, we

introduced a module to build and extract the relationship of acronymy, for example: The acronym for ”Data Encryption Standard” is DES. The goal of these relationships in ontology was to organize the concepts hierarchically and since the approach is cooperative, the ontologist can intervene in this phase to validate the concepts and the semantic relations.

Step 2: Execution of the process of identification of the concepts

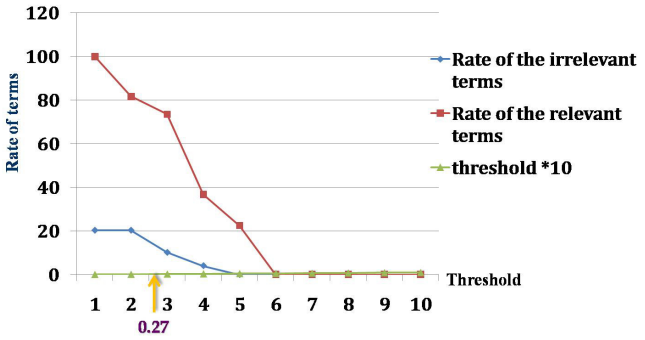

Fig. 2. Similarity estimation.

In the end of this process, we have constructed two sets (the terms known and unknown by WN1.2) and for identified the relevant and irrelevant terms, we have effected several tests for choose the minimum threshold which allow to identifier the maximum of the relevant terms. The selection curves of the threshold are clearly related the similarity of the relevant, irrelevant terms and the variation of the thresholds. We have chose the threshold where the number of the relevant terms is more important and the number of the irrelevant terms is negligible; the selected threshold is 0.27. From the results obtained, the core ontology contains 76 relevant terms. From the results obtained, the core ontology contains 76 relevant terms. Then, to enrich the ontology using these words core in the disambiguation step. The desambigisation of candidate terms allows to identify the sense and the projection of these terms on WN1.2 allows to define their synonyms and their father and relation with the other terms of the corpus. For the terms unknown by the WN, the WSDA applies the rule 3. for example, the term “ public key”. If Heat”public key” ϵ S (i.e.,key) then add this terms to S and if the term ”program” ϵ S and the term ”RIPEM” is in subsumption relation with the term “program”, then adding the term “RIPEM” to S for example ”RIPEM is a program which performs Privacy Enhanced Mail (PEM) using the cryptographic techniques of RSA and DES...RSA is a crypto system which is public-key...” S = S + {RIPEM }, S = S + {PEM} and S = S + {RSA}. Concepts represented by the dotted rectangles are the new

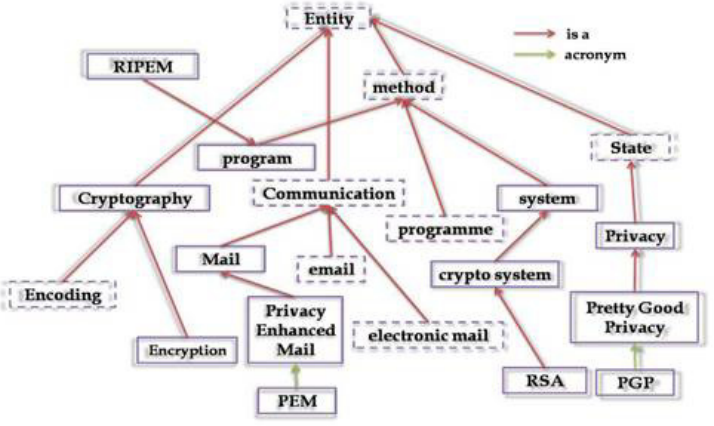

Fig. 3. A part of the ontology.

concepts added automatically from WN1.2 or the concepts added by the ontologist. In the end of this step, we have enriched the ontology with 107 concepts. Then, for each given term in output of this process, WSDA created the SSA

agents and for the terms that represent common nodes, it creates the LA agent.

Step 3: Execution of the process of hierarchical of the concepts

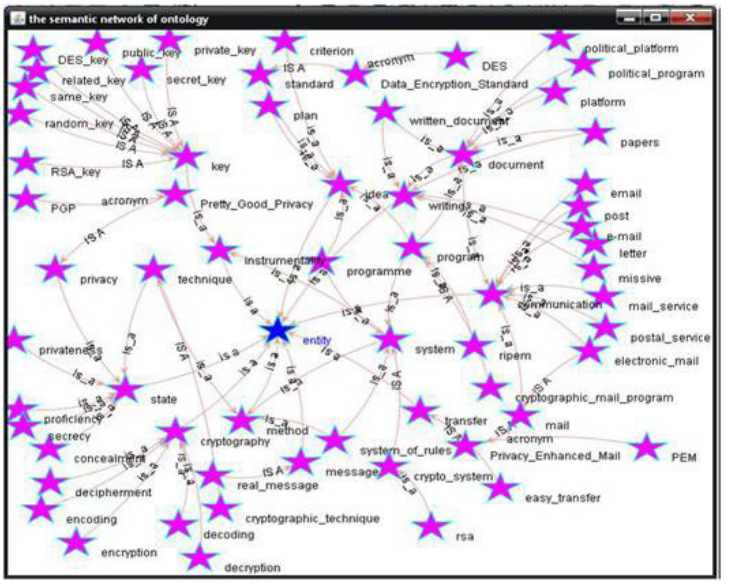

Fig. 4. Screen capture of the DEO tool.

Given a term *mail*, what is the best way of joining it with their unique similar term? . The agent SSA communicates with LA agent whose goal is to build the hierarchy. The SSA agent send the message *send*(*mail*) where it asked, by this message, the AL agent to send him the father of the term *mail* and LA agent answers by the message Answer(message, communication),Then the Assi agent saves the tuples *(mail, message , communication)* in Temp file and the LA agent annotates the relationship. Then the SSA_i and SSA_j agent destroyed dynamically; The process stops when no SSA agent is available. In the end a concept entity is created where all the pivot concepts of these hierarchies constructed are directed to this entity concept. For example $cryptographic\_technique \xrightarrow{IS\_A} technique \xrightarrow{IS\_A} method \xrightarrow{IS\_A} entity$. The first link is detected by using head-expansion network and the second link is added from the ontology WN1.2. It

Table 1. The characteristics of DEO and other methods of construction of ontologies from texts

Methodologies	**Approaches**	**Tools**	**Type of analysis**	**Learning from**	**MAS**
ASIUM	Structural	Lexico-syntactic filters	Semantic	Text	No
DYNAMO	Structural	Syntex	Semantic	Text & experts	Yes
OntoLearn	Mixed	Not specified	Statistical & conceptual	Text	No
Text2Onto	Mixed	WordNet	Statistical & conceptual	Text	No
DEO	Structural	TreeTager, WN1.2 & Head-Expansion link	Semantic	Text & experts	Yes

is extremely difficult to assess the relevance of a method or methodology, as there is yet no consensus or rules for evaluation, but experimentation is naturally the appropriate way to do. In the Table 1, we have mentioned some characteristics of methods and methodologies to build ontologies by learning. In the context of comparison with DEO, we considered: The approaches used to extract terms can be structural approaches, non-structural approaches such

as quantitative and statistical approaches and hybrid approach. The tools used for the extraction of terminology, currently, there are several tools for the terms and relationships extraction, they are organized into four groups: tools for extract concordances such as Yakwa, Sato, tools for terms extraction like Lexter (Syntex), Nomino, tools for relations extraction such as Chameleon, Likes, Prometheus and tools for syntactic categorization of words: Cordial, TreeTager. Type of analysis used to identify concepts (semantic analysis or statistical analysis). Learning is the acquisition of knowledge from the text, dictionary, knowledge bases or directly from experts. Using of MAS we can say that ontology is an interesting type of knowledge modelling in which concepts are autonomous entities, relationships are the interactions between these entities and structures of these knowledges is kind of coordination/control. In this time, we can see ontology as an application in the field of artificial intelligence. While several agents share common resources and communicate with each other to make the structure of the ontology dynamics.

6. Conclusion and perspectives

Our work is situated in the field of ontology engineering, of the Semantic Web and the MAS. Our goal was to automate the process of ontology construction. The result of this work is a new original approach ontology construction named DEO. It is a tool for building ontologies from text by cooperative learning. For the extraction of relevant information, it uses the mechanism for extracting relations of dynamo and the more it uses the terminology WN1.2 to identify the concepts and the relationships. Similarly, it exploits a module of interaction system-user to modify or validate these concepts and this relation. The latter is managed by software agents, the more DEO uses a backup module to record the modifications of ontologists for reuse. The experiments conducted until now have focused on a collection of relatively modest size text. But it is important to evaluate our approach on more realistic corpus sizes. In future, we should test DEO prototype In a dynamic environment (dynamic corpus) Where the collections of documents can be changed; this implies keep updated ontology which is a task until here underestimated. For this reason we think that the use of an agent can remedy these deficits, because it can discover change of the Environment by using these perceptions.

References

1. Kvin Ottens, Marie-Pierre Gleizes, and Pierre Glize. A Multi-Agent System for Building Dynamic Ontologies, AAMAS, May 1418, 2007, Honolulu, Hawaii, USA.; p. 51-93.
2. Alexander Maedche. Ontology learning for the Semantic Web, Boston Kluwer Academic Publishers 2003.
3. Jacques Ferber. Les Systmes Multi Agents: Vers une intelligence collective, interEditions, Paris; 1995.
4. Antonio De Nicola, Michele Missikoff, Navigli Roberto. "A Software Engineering Approach to Ontology Building", Information Systems (Elsevier) 34 (2): 258275; 2009.
5. Yoshinobu Kitamura, and Riichiro Miezouguchi. Ontology-based description of functional design knowledge and its use in a functional way server, Expert Systems with Application,2003, 24(2), 153-166.
6. Neches Robert, Fikes Richard, Finin Tom, Gruber Thomas, Patil Ramesh, Senator Tod, Swartout William. Enabling technology for knowledge sharing,AI Magazine,Winter 1991,36
7. Gayo Diallo. Une Architecture Base dOntologies pour la Gestion Unifie des Donnes Structures et non Structures", PHD thesis, University of J. FOURIER, Grenoble, France, december 2006.
8. Marie-Aude Aufaure. Ontologies et Fouille de Donnes pour un Web plus smantique: application la construction et lvolution dontologies, et la personnalisation web", EcolIA, March 2008.
9. Alexander MAEDCHE, Steffen STAAB. Mining Ontologies from Text. 12th International Conference on Knowledge Engineering and Knowledge Management, LNAI Springer, Juan-les-Pins (France), 2000, p. 189-202.
10. Thierry Poibeau, Dominique Dutoit , Sophie Bizouard. "valuer lacquisition semi-automatique de classes smantiques". TALN 2002, Nancy, june 2002, p. 24-27.
11. Zied Sellami, Nathalie Aussenac-Gilles, Marie-Pierre Gleizes. Vers un outil de co-construction dontologies partir de textes laide dun systme multi-agent adaptatif, Technique et Science Informatiques, Herms Science Publications, 2011.
12. http://taln09.blogspot.com/2009/02/etiquetage-morpho-syntaxique-et.html
13. WN 1.2, http://www.rocketdownload.com/program/word-net-8879.html.
14. http://fr.wikipedia.org/wiki/Polys%C3%A9mie
15. Wu Z, Palmer M, "Verb Semantics and Lexical Selection", Proceedings of the 32(nd) Annual Meetings of the Associations for Computational Linguistics, p.133-138, (1994).
16. Lin D,. "An information-theoretic definition of similarity" In Proceedings of 15(th) International Conference On Machine Learning, (1998).
17. http://kaon2.semanticweb.org/ Universit de Stanford. The Protg Ontology Editor and Knowledge Acquisition System.
18. http://protege.stanford.edu/
19. http://people.csail.mit.edu/jrennie/20Newsgroups/