Procedure for the Contextual, Textual and Ontological Construction of Specialized Knowledge Bases

François Achille Djontu Tajouo, Thierry Noulamo, and Jean-Pierre Lienou

Abstract — Information Retrieval (IR) from data sources such as the web, databases, sensors, etc., and structuring it through ontologies is an ambitious research field that leads to the reduction of information search time, improvement of the quality of found information and the efficient decision making. In this work, we have proposed a system for extracting information from several data sources to enrich an ontology in the field of plant pathology. The objective of this work is to develop a knowledge base in the field of plant pathology, by structuring the data present in various data sources accessible mainly through the web. To do this, we first propose a modular development approach of a domain ontology. The construction of the ontology at first uses the text-based construction technique and then, we use the generic architecture of a web data extraction system technique from literature to propose a data extraction system for the enrichment of the different ontology modules of the domain. For each subsystem of our extraction system, we have developed the algorithms to be implemented.

Index Terms — Information Retrieval, Knowledge Base, Merge ontologies, Modularization, Ontology Module, Ontology.

I. INTRODUCTION

The web is the main source of data that is open and accessible almost to all today. It contains a variety of data which are structured (Databases) or unstructured (Web Pages), which require many hours of research with current tools and which do not always give precise results. However, decision making in dynamic fields requires the timely availability of accurate and quality information.

Furthermore, ontologies allow the structuring of concepts in a domain and are used to build knowledge bases. However, in spite of the fact that most ontology development methods depend on the domain being modelled and the desired application of the ontology, the construction method using texts remains dominant in the literature [2]-[5], since it allows us to free oneself from the presence of experts in the domain. These methods do not generally take into account the extent of the domain which can at a given time, because of the large volume of concepts, pose the problem of limited resources and the limitation of human capacities in the updating of ontology concepts and reasoning by the inference engine. However, the modularization of ontologies not only allows projections to be made on the sub-domains of the knowledge domain modelled for certain applications, but also proves to be a solution to facilitate the updating of the ontology [6], [7].

Faced with this problem, we propose in this work a modular construction approach of domain ontology which is inspired by the paradigm "Divide and conquer" and which exploits the method of construction from the texts for the enrichment of the ontology.

Search engines such as Google rely mostly on indexing and keyword-based search methods. Although recently, with the emergence of the semantic web, we have seen results that suggest that the semantics of the query are taken into account when searching for information. However, this approach remains imprecise and time-consuming. This imprecision is due to the fact that the context of the pages is not taken into account when extracting information [1]. This is what inspired the integration of ontologies in the information retrieval process [1], [8]-[10].

In this work, we are interested in the extraction of information from several data sources, taking into account the research context, and then the structuring of the extracted data by the ontological modules to obtain a knowledge base in the chosen field. This approach is original because it integrates into the information retrieval process the ontologies that structure the knowledge of a domain and allow to take into account the context of the research, and the extracted data are used to enrich a knowledge base.

This paper, in its first section, situates the subject in the literature by presenting the field of information retrieval, then, the interest of building ontologies from the texts and the justification of the choice of the modular approach for the development of the ontology of the domain. The second section, explains the methodology adopted, presenting the architecture of our extraction system, which is a reprise of the generic architecture proposed by Espinasse B. and al. The third section, entitled results and discussions, presents the importance of the choice of the domain, the ontology modules obtained, and the algorithms simulated in our extraction system. This work concludes by giving openings for future work in the field.

II. INFORMATION RETRIEVAL AND CONSTRUCTION OF ONTOLOGIES

A. Review Stage

In this section, we present the work carried out in the field of information retrieval. Since information is extracted for a

Published on February 2, 2021.

François Achille Djontu Tajouo, Department of Mathematics and Computer Science, University of Dschang, Cameroon.
(e-mail: djontu.tajouo@gmail.com.)

Thierry Noulamo, Department of Computer Engineering, IUT FOTSO VICTOR, University of Dschang, Cameroon.
(e-mail: thierry.noulamo@gmail.com).

Jean-Pierre Lienou, College of Technology, The University of Bamenda, Cameroon.
(e-mail: lienou@gmail.com)

specific application and in this case for the construction of a domain ontology, work on the construction of ontologies from texts will be the subject of the second part of this section.

B. Information Retrieval

Information retrieval is the process of creating knowledge from structured information (relational databases, XML) or unstructured information (texts, documents, images). The result must be in a computer-readable format.

For Ait, Radi and al. [11], Information Retrieval (IR) is defined as the process of identifying relevant information, where the criteria for relevance are defined in the form of templates to be filled in. It describes an Information Retrieval process which is divided into four steps which are as follows: Text pre-processing and transformation; data selection and reduction; data mining; analysis, interpretation and validation of results.

In order to obtain accurate information and improve search time, domain ontologies are increasingly integrated into the extraction process. That is why Gabin, Personeni [10], was interested in ways to use biomedical ontologies to extend the possibilities of the data mining process. It uses pattern structures, an extension of formal concept analysis (CFA), to extract association rules between adverse drug reactions to certain drugs or classes of drugs in groups of patients.

Abirami A. et al. [9], proposes a model for storing the information available on web pages in an organized and structured form in RDF format. In its system, keywords are given to the search engine that retrieves HTML (HypeText Markup Language) pages. The system learns the structure of these pages and a pointer is moved to the section where the necessary data or information is presented. Upstream, the vocabulary of the domain is described by RDF graphs. A correspondence between the user's query and the ontology is established and then the result is refined and returned to the user. The RDF graph is used to store the data from the Html pages and the query is performed by SPARQL and JENA. It shows that the time needed to infer new knowledge is also minimal when semantic web technologies are used for the development of these applications, compared to the manual approach.

Espinasse B. and al. [1], proposes a generic architecture based on agents and ontologies for the collection of information on restricted domains of the Web. He insisted on the fact that taking into account the context in the search for information is possible and leads to more relevant collections before proposing a cooperative collection of information based on software agents and ontologies. For the implementation of its solution, a generic software architecture, AGATHE (Agents information GATHERing), implementing this type of collection and allowing the development of collection systems relating to one or several domains, is presented in detail.

The extracted data necessarily have a purpose, and in the case of species, they will be structured by a domain ontology and allow the enrichment of a knowledge base.

C. Building Ontologies from Texts

Several existing ontologies are constructed by text analysis techniques. In fact, in order to free themselves from the

presence of experts in the domain, the texts prove to be the bearers of consensual information shared by experts in a given domain. The means used is to start from existing elements in the field such as textual corpora, taxonomies, fragments of pre-existing ontologies, database schemas, etc. and use them as a priori knowledge and enrich the ontology progressively.

There are several methods and tools for building ontologies from texts and a comparative study has been carried out by Assia Amarir and al. [5]. These approaches to building ontologies from texts can be grouped into two groups, namely:

- Terminological approaches that use Automatic Language Processing (ALP) tools to retrieve the remarkable elements of the text and require the intervention of the ontologist to formalize the ontology from the concepts and relationships extracted. Examples include: the TERMINAE tool [4]; Text-To-Onto [3]; Upery [13].

- Non terminological approaches which tend to use algorithms and the philosophy of machine learning, to make an almost automatic conceptualization from texts without going through a phase of terminological extraction and validation. Examples include: Text2Onto [2]; OntoGen [14]; OntoLearn [15].

D. Modular Construction of Ontologies

The construction of domain ontology by considering the domain of knowledge in its globalism requires both the resources for calculation during updates, but also the human resource to validate the concepts and their meaning. Faced with these limitations, a modular approach to the development of domain ontologies and the use of operations such as fusion and alignment proves to be a better solution to better dominate the domain to be formalized.

For Maria Keet [15], modularization refers to the division of an ontology to help simulate large ontologies. It is necessary when it is necessary to hide or remove knowledge that is not required for the application in question, or to divide so that modules can be worked on separately.

For Zubeida and al. [6], modularity in ontology engineering is applied as a solution for dealing with information overload for both machines and humans, as it facilitates computational tasks and simplifies the understanding and interpretation of knowledge by offering smaller subsets of an ontology. Furthermore, they define an ontology module M as a subset of a source ontology O , $M \subset O$, either by abstraction, removal or decomposition, where module M is an ontology existing in a set of modules such that, when combined, they constitute a larger ontology. Module M is created for a certain use case U . It is this second part of the definition given by Zubeida and al [6] which is implemented in this work.

Definition 2.1

An ontology module is a formal, conceptual and consensual modelling of a sub-domain of knowledge that is part of a larger knowledge domain.

The subdivision of the domain into sub-domains is intended to facilitate the manipulation of knowledge by reducing the field of action of the algorithms.

III. METHODOLOGY FOR BUILDING THE KNOWLEDGE BASE

A knowledge base is the combination of two levels of concepts, namely the terminology level (TBox) and the assertion level (ABox).

Our approach consists first of all in manually constructing, using the Protégé Editor, the cores of the ontology modules that model the sub-domains of the global target domain. The level implemented here is the terminology level. These ontology module cores will be used to structure user queries to direct searches in data sources. The data extracted from the data sources will be used, on the one hand, to extend the kernels of ontological modules, and, on the other hand, to enrich the assertion level and, consequently, the development of the knowledge base.

The general architecture of our system is given by the Fig. 1. This architecture is derived from the architecture proposed by Espinasse B. and al. [1] and brings in addition the enrichment of the ontologies from the data extracted from the different sources.

When the user needs some information, he makes the request to the US (User Subsystem) through an interface provided for this purpose (1a). If the knowledge base contains the answers to the user's query, then the information is returned (case of searching for information that has already been constructed). Otherwise (1b) the query is enriched using the concepts present in the ontology (Terminology level) (2a). The user's query to classic search engines often does not take into account the underlying consensual concepts of the domain. The role of the ontology is therefore to provide the consensual concepts but also the relationships between these concepts. The Wordnet library is also used to find all the

synonyms of the words describing the concepts (2b).

Once the query is enriched, it is transmitted to the Search Subsystem (SS) (3a). The SS will browse data sources (web pages, databases, data warehouses, etc.), by querying existing search engines, particularly Google (4a, b). A list of URLs is then transmitted to the Extraction Subsystem (ES), more precisely to the extraction cluster corresponding to the knowledge sub-domain being processed (5). If necessary, recommendations are sent by the cluster in question to other extraction clusters, in order to propose pages that may potentially be of interest to them (6). From the pages obtained, the interested cluster extracts the relevant concepts and the relations between these concepts, in order to populate the initial ontology, thus updating the knowledge base and facilitating the user's future research.

In order to keep the knowledge base up to date and respond to the user without the need to process the query, we have included in the search sub-system, scientific monitoring tools, that will take care of retrieving in real time (4b), links to new publications in the target domain and returning the Urls of the pages to the retrieval sub-system. The scientific monitoring tools taking into account the context of the ontology (3b).

The role of the mapping module is to make data sources transparent and to standardize queries to them. It is the interface between the data sources and the search subsystem. The extracted data is structured as an XML file and the Uri (Uniform Resources Identification) is sent to the ES.

The modular development approach to domain ontologies that we advocate in this work is of particular interest. Indeed, extraction clusters are not only built on the basis of specific ontology modules, but enrichment is also done by ontology module.

Fig. 1. Data extraction system and ontology enrichment.

IV. RESULTS AND DISCUSSIONS

A. Example of Ontology Construction in Plant Pathology

The construction of a domain ontology is not always obvious, especially for a domain as vast as phytopathology, and requires expert knowledge. It should be noted that phytopathology includes knowledge of plants, of the different diseases that can attack these plants, depending on the environmental conditions and the presence of pathogens in the environment, but also knowledge of treatments and treatment processes, which vary not only according to the seasons, the environment, the type of plants and soil, and the

environment where they are applied.

To gain a better understanding of this domain, we have subdivided it into two sub-domains: "plants and diseases" and "diseases and treatments". For each sub-domain, we have developed a core ontology module using the Protégé 5.5.0 editor, as shown in Fig. 5 and 6. To do this, we followed the following steps:

- Step 1: Understanding the scope and purpose of the ontology

This step allowed us to delineate the contours of the ontology in terms of the concept to be taken into account. We have defined the purpose of our ontologies and the questions that the knowledge base is likely to answer.

- *Step 2: Collecting and analyzing domain information*

To talk about a disease in a given plant, there is a set of concepts to be taken into account, and this set of concepts is structured on the vertices of a triangle called the disease triangle. At one vertex of the triangle, you have the host plant, on the other two vertices you have the environmental conditions at each of the other two vertices the pathogen responsible for the disease as shown in Fig. 2.

With this set of information, we moved to the conceptualization stage with UML.

- *Step 3: Conceptualization of the ontology*

The conceptualization of the ontology modules was done using a UML class diagram. As shown in Fig. 3 and 4.

- *Step 4: Formalization of the ontology*

This phase consists of editing the ontology in an ontology development environment and the one we have chosen here is the Protégé 5.5.0 environment. The exploitation of all the

previous data and the conceptualization allowed us to obtain the first ontology modules as shown in Fig. 5 and 6.

These modules can, depending on the situation, be merged to have a larger ontology on plant diseases and their treatments using Protégé tools.

Fig. 2. The Disease Triangle.

Fig. 2. Conceptualization of plant disease ontology.

Fig. 3. Conceptualization of plant disease ontology.

Fig. 4. Graph of the ontology module on plant diseases.

Fig. 5. Graph of the ontology module on the treatment of diseases.

B. Presentation of algorithms

For each sub-system of our architecture, we have developed algorithms that will be implemented to build our system.

Since the user's words may not be the formal word for the concept, we use Wordnet to find synonyms and concepts from the ontology. The words are then replaced in the user's expression and the query thus enriched is sent to the RS who will submit it to the data sources.

The function of enriching the user's query with the concept structure of the ontology is as described on Algorithm 1.

Once the query has been formulated, it is sent to the search subsystem for data mining. The search function for the web pages corresponding to the concepts present in the list of concepts sent by the US is presented as described on Algorithm 2.

This sub-system will apply web scraping to extract the content of the web pages found. Note that web scraping (sometimes called harvesting) is a technique for extracting content from websites, via a script or program, in order to transform it to allow its use in another context.

The function for extracting data from web pages sent by the SS is as described on Algorithm 3.

The text fragments provided by the previous function will be used to enrich the ontology modules using one of the ontologies building techniques from the texts presented as described on Algorithm 4.

The function of ontology enrichment is given below.

This approach to building the knowledge base can easily be extended to other areas of knowledge. Indeed, and as mentioned in the introduction, the web in particular contains a lot of knowledge and the construction of ontological modules and the adaptation of extraction clusters are sufficient to build a knowledge base in a chosen domain.

Algorithm 1: Enrich Request (String)

```

Function: enrichRequest(string)
1  dic ← convertStrgWord(string)           /*Convert the string to a word and store it
                                           in a dictionary*/
2  load(ontology)                          /*Loading the ontology */
3  load(wordnet)                          /*Loading the wordnet library*/
4  For word ∈ dic:
5  listConcept ← ci ∈ CSuch                /* C set of ontology concepts */
   as
   ci ∈ synset(word) or ci ∈ Domain(word) ∪ Range(word)
6  return (listConcept).

```

Algorithm 2: Research Webpage(listConcept)

```

Function: ResearchWebPage(listConcept) :
1  links[]                                /*empty list that will receive the links to
                                           the pages*/
2  For concept ∈ listConcept
3  req ← concept                          /*Giving the concept to mapping module */
4  link ← getUrl(req)                     /*We extract the links from the previous
                                           page*/
5  links[] ← link
6  return (links).

```

Algorithm 3: Data Extraction(links)

```

Function: dataExtraction(links) :
1  For link ∈ links:
2  page ← open(link)                     /*Make the query on the link*/
3  content ← parser(page)                 /*Select the content of the page by eliminating
                                           tags and
                                           formatting.*/
4  write(content, file)                   /* Write the content of the page to a file */
5  return (file).

```

Algorithm 4: Enrich Onto (File)

```

Function: enrichiOnto(file) :
1  load(onto)                            /*Loading the ontology */
2  For ligne ∈ file:
3  wordList ← convertStrgWord(ligne)
4  For word ∈ wordList:
5  If word ∈ CSynset(ci ∈ C) Then          /*C is the signature of ontology.*/
6  add(word, C)                          /*Add the word to the set of ontology concepts*/
7  Else If word ∈ CSynset(ci ∈ C) Then
8  add(word, C, equivalent)              /*adds the word to the signature as the
                                           equivalent of an ontology concept. */
9  return (onto).

```

V. CONCLUSION FURTHER WORKS

In this paper, we have proposed an approach to build a knowledge base using ontologies and a system of context-sensitive textual data extraction. In order to do so, we have situated the work in the literature, recalling some works done both in the fields of Information Retrieval and ontology construction. After justifying the choice of the modular approach to the development of domain ontologies, we proposed an architecture for the construction of a knowledge base and developed the algorithms to enriching the user's query, the search function for the web pages corresponding to the concepts, the web pages data extraction function and the ontology enrichment function.

This work does not pretend to have gone around the issue. In future work, we will focus on the technique of enriching ontological modules, in particular integrating artificial intelligence algorithms, to increase the independence of the system and implemented previous algorithms with python language.

REFERENCES

- [1] Espinasse, B. and Fournier, S. and Freitas, F. (2007) "AGATHE : une architecture générique à base d'agents et d'ontologies pour la collecte d'information sur domaines restreints du Web", CORIA 2007, Conference Paper, pp 367-383
- [2] Philipp, Cimiano & Johanna, Völker, (2005) "Text2Onto - a framework for ontology learning and data-driven change discovery", Proceedings of the 10th International Conference on Applications of Natural Language to Information Systems (NLDB), Vol. 3513, pp 227-238.
- [3] Alexander, Maedche & Steffen, Staab (2000), "Discovering conceptual relations from text", Dans ECAI 2000, Proceedings of the 14th European Conference on Artificial Intelligence, IOS Press, Amsterdam, pp 16-17.
- [4] Brigitte, Biebow & Sylvie, Szulman (1999), "TERMINAE: A linguistics-based tool for the building of a domain ontology", Knowledge Acquisition, Modeling and Management, N° 1621, pp 49-66.
- [5] Assia, Amarir, El-Habib & El-Houssine, (2014) "Les méthodes de construction d'ontologie à partir de texte", La deuxième journée sur les Technologies d'Information et de Modélisation TIM'14. Faculte des sciences Ben M'sik; Université Hassan II.
- [6] Zubeida, Casmod & Maria, Keet, (2015) "An empirically-based framework for ontology modularization", Applied Ontology, vol. 10, no. 3-4, pp. 171-195.
- [7] D'Aquin, M. & Schlicht, A. & Stuckenschmidt, H. & Sabou, M. (2009) "Criteria and evaluation for ontology modularization techniques. In Modular Ontologies: Concepts, Theories and Techniques for Knowledge Modularization", Part of the Lecture Notes in Computer Science book series (LNCS, volume 5445), pp 67-89.
- [8] Mouhamadou, S. & Moussa, L. & Cheikh, T. & Fatou, K. (2011) "Etat de l'art sur ontologies et extraction de connaissances", Conference: Colloque National sur la recherche en Informatique et ses applications, At Senegal.
- [9] Abirami, A. and Askarunisa, (2013) "A semantic based approach for knowledge discovery and acquisition from multiple web pages using ontologies", International Journal of Web & Semantic Technology (IJWesT) Vol.4, No.3.
- [10] Gabin, Personeni, (2018) "Apport des ontologies de domaine pour l'extraction de connaissances à partir de données biomédicales", Thèse, Université de Lorraine.
- [11] Ait, Radi and Iggui, Taous, (2013) "Extraction d'information à base d'ontologie pour la construction d'un profil utilisateur.", Université A. Mira Béjaïa, Mémoire de Master.
- [12] Didier, Bourigault, (2002) "Upery : un outil d'analyse distributionnelle étendue pour la construction d'ontologies à partir de corpus", Actes des 9emes journées sur le Traitement Automatique des Langues Naturelles, pp 75-84.
- [13] Blaz, Fortuna, Marko, Grobelnik & Dunja, Mladenic (2006) "Semi-automatic data driven ontology construction system", Proceedings of

the 9th International multi-conference Information Society IS-2006, Ljubljana, Slovenia, pp 3-4.

- [14] Paola, VELARDI, Roberto, NAVIGLI, Andrea, CUCCHIARELLI and NERI, F., (2005) "Evaluation of ontolearn, a methodology for automatic learning of domain ontologies", Ontology Learning from Text: Methods, Evaluation and Applications. IOS Press, Amsterdam.
- [15] Maria, Keet (2020) "An Introduction to Ontology Engineering", EBook, Available on <https://people.cs.uct.ac.za/~mkeet/files/OEbook.pdf>.

François. A. Djontu Tajouo, Software Engineer, starts research in information extraction using semantic Web technologies.

Dr. Thierry Noulamo, PhD. Lecturer-Researcher, Department of Computer Engineering, IUT Bandjoun, The University of Dschang. Dr NOULAMO obtained his MSc (DEA) in Computer Engineering in the Faculty of Science (The University of Yaounde I) and his PhD in University of Yaounde I (Cameroon). He joined the University of Dschang in 2008. His Research fields are Model Driven Engineering applied in complex systems, and the use of various artificial intelligent techniques in diagnostic of complex systems.

Dr. Jean-Pierre Lienou, PhD. Lecturer-Researcher, Head of Department of Computer Engineering, College of Technology, The University of Bamenda. Dr Lienou obtained his MSc in System Engineering in Kiev Polytechnic Institute (National Technical University of Ukraine) and his PhD in University of Yaounde I (Cameroon). Worked as Maintenance Engineer at Labotech Medical and was in charge of Medical imaging equipment before joining the University of Dschang. His Research fields are Method Engineering applied in control systems, Multi Agent Systems applied in power electric systems and the use of various artificial intelligent techniques in diagnostic of complex systems.