Journal of Theoretical and Applied Information Technology
15th January 2018. Vol.96. No 1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

AUTOMATIC ARABIC ONTOLOGY GENERATION FOR THE ANIMAL KINGDOM (NAAO)

[1] DALIA FADL, [2]SAFIA ABBAS, [3]MOSTAFA AREF
Faculty of Computer and Information Science, Ain Shams University, Egypt

E-mail: [1]dalia_sayed_43@hotmail.com, [2] safia_abbas@yahoo.com, [3]mostafa.m.aref@gmail.com

ABSTRACT

Ontology is the gateway for the interaction between human and the machines because it facilitates the gaining of the information for the end users. The semantic web terms and vocabulary are considered as form of ontology. So the improvement of the ontology helps in the semantic web progration. Arabic ontology is a critical natural language processing field; it helps to enrich the Arabic language resources. Although, the searchers started to focus their efforts creating ontologies in Arabic language, the Arabic terms and fields have not yet been covered. This paper introduces an innovative framework, annotated as NAAO (Novel Automatic Arabic Ontology), which automates the ontology generation process from XML documents. The novelty of NAAO resides in generating the Arabic ontology, in the form of XML graph schema (XSG), from semi-structured data (XML documents associated with graph schema). The automation process entails four main phases (extrication, XML schema parsing, ontology generation and refinement and evaluation). Extrication phase, where all the data required for the ontology generation is extricated from the information source. XML schema parsing phase, where the extracted data is going to be investigated and the hierarchical analysis to identify all the parts of the ontology is going to be created. Ontology generation phase deal with the production of the Arabic ontology (classes, elements, and relations). Finally, refinement and evaluation phase that deals with improvements of the Arabic ontology. Moreover, the paper comprises versatile examples for an automatic generation to Arabic ontologies regarding the animal kingdom. Such as the invertebrate's ontology, in which the XSG is automatically extracted using 103 XML documents, 320 concepts (classes), 603 elements and 783 relations. Finally, NAAO enjoys certain advantages than others since it shows an automatically generate Arabic ontology from semi-structured data (XML file). It provides mapping connections to ensure the right relationships with the classes and the elements to its corresponding nodes in the XML documents and supplies huge ontology contents (concepts elements and relations). Moreover, NAAO offers an automatic evaluation of the generated Arabic ontology using cosine similarity measurements.

Keywords: *Arabic Ontology, Ontology Learning, Ontology Generation, XML Schema, Evaluation*

1. INTRODUCTION

Semantic web allows the existence of a wide range of web information and services to be accessible by humans and automated tools. Ontology introduces a machine-readable formation of basic concepts in a domain and in which it extracts the relation between them. Ontology generation helps to limit the complexity and organize the information on the web to be more intelligent. It improves the searching results returning the meaning and literature meaning [1,2,3]. The semantic web depends on the formation of ontologies to structure data for more intelligent and comprehensive generation of the web. The vision of the next generation of the web (semantic web) has spread widely so the need for the ontology became a must to act as a bridge between computers and human natural language.

The rise of linguistic ontologies is an aftereffect of two simultaneous subjects: information structuring and knowledge representation. Such topics facilitate the user exploitation. The need for linguistic data is critical in all research fields that concerned with the organization and retrieval of information. The Arabic Ontology is a formal representation of the concepts that the Arabic terms convey [2]. For each term in the Arabic language, a set of its similar words are identified, and semantic

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

relationships between all concepts are introduced [2]. For simplicity, the Arabic ontology is a tree of the meanings of the Arabic terms. It can be seen as a dictionary [3]. However, its relationships are well-formalized, and glosses follow strict formulation and ontological rules. Specification and conceptualization are the two most critical phases of ontology development. Specification phase aims to gain knowledge about the domain. While conceptualization phase expects to sort out and structure this information using external representations that are autonomous of the execution languages and environments [4].

A lot of efforts have been done in the ontology generation. These ontologies have covered lots of fields and use different techniques [6, 7, 9, and 12]. The Arabic ontology generation still needs a lot of effort and extensive work to improve the construction of the ontology [7]. The manipulation Arabic language is considered a difficult process, since many features, that may obstruct the development of Arabic ontology, are entailed. Among these features, is the complex morphological, grammatical and semantics [7]. Moreover, Arabic language is characterized by ambiguity due to the vowelization (different words have similar pronunciation with a different meaning), Polysemous (a same word with a different meaning) [7, 8].

There are three fundamental ways to create ontology (manually, semi-automatically and automatically) [9]. The manual ontology generation is to build ontology from "scratch", in which classes, relations, and instants are defined. The automatically and semi-automatically is to create ontology using ontology learning techniques. The goal of ontology learning is to automatically extract relevant concepts and relations from the given input sets to form Ontology [4]. At Semi-automatic Ontology creation; the approach depends on the way that the source data have equivalent structures and usefulness as the hidden information. The automatic ontology generation is a very promising field of ontology learning because it presents more accurate results for the process of the generations and makes it easier[4].

Accordingly, NAAO aims to generate criteria for designing automatic Ontology for the Arabic language. Describe the Role of Ontologies in supporting knowledge sharing activities. Take an engineering perspective on the development of ontologies in the Arabic language. Design and Develop a prototype model for an automatic ontology framework for the Arabic language. Automate the ontology evaluation process to decrease the time consumption for the evaluation process.

The Theoretical implications of generating automatic Arabic ontology ease the access to multicultural and multilingual resources. Simplifying and improving in the way the human-computer interaction and communication for Arabic language sharing and reusing. It is generally accepted that building an Arabic ontology is a difficult task, and this task could be greatly facilitated as much as possible to reuse and modify ontologies created by others. The high manual cost of ontology generation makes the creation of the ontology automatically is a very important and useful task. Practical implications of the generation of the framework it helps coping with the growth of the Arabic text over the internet. This exponential growth needs a way to structure the vocabularies, and formal taxonomies, the generation of the Arabic ontology helps in this structuring.

We have designed and implemented an innovative framework that automates the ontology generation process. The novelty resides in generating the Arabic ontology, in the form of XML graph schema (XSG), from semi-structured data (XML documents associated with graph schema). The framework makes refinement to the generated ontology using tree-based mining technique. It also provides an automatic evaluation of the generated Arabic ontology using cosine similarity measures. The generated Arabic ontology includes a wide range of vertebrate and invertebrate animals.

This paper is organized as follows. Section 1 gives the research motivations, objectives and implications. Section 2 presents the Arabic ontology generation related work. The Arabic ontology architecture and its four phases are described in Section 3. Section 4 presents the system implementation, input documents, and the generated ontology. Also, it shows the Arabic ontology output analysis. Finally, the paper is concluded in section 5 by highlighting the main features of NAAO. Also, Section 5 gives the contributions of this research work.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2. RELATED WORK

Many systems have created ontology manually such as M.Jarrar in [29], where an Arabic linguistic ontology has been developed and implemented. The top level of the Arabic ontology was built manually based on DOLCE and SUMO upper-level ontologies. 420 concepts of the ontology have been evaluated, and the remaining concepts have been completed. Whereas, H. Aliane, et.al, in [32] presented Al –Khalil project for building Arabic infrastructure ontology. The main interest of this work is a linguistic ontology that is founded on Arabic traditional grammar. The project contraction consists of two steps; the first step is bootstrapping manually the ontology by choosing the linguistic concepts and relating them to the concepts in gold. The second is using an automatic extraction algorithm to extract new concepts from linguistic texts to enrich the ontology. Hegazy, et.al, [9] represented Arabic financial accounting knowledge in the computer domain using ontologies. It provides a methodology for the creation of ontologies based on declarative knowledge representation systems. A narrow accounting vocabulary was used in the system. The ontology has been implemented manually using protégé-4.3[9].

Ontology learning refers to the automatic or the semi-automatic construction of the ontology. The ontology learning process is an efficient way to accelerate the process of ontology generation to cope with the web information growth. Maryam Hazman, et.al, [5] generated a semi-automatic ontology. The system takes the root concept as an input, analysing all information documents' heading structure, extracting concepts from the headings and building a taxonomical ontology. After then, ontology from heading titles is generated based on a set of web documents using information that exists in the title's text as well as the HTML structure [5]. R. Ghawi, et.al, [7] generated a semi-automatic ontology. It is a proposed tool, called X2OWL that aims to build OWL ontology from an XML data source. This method is based on XML schema to automatically generate the ontology and an arrangement of mapping bridges between the elements of the XML information source and the created ontology. These mapping bridges contribute to query translation between OWL and XML [7]. Bedr-eddine, et.al, [10] presented a system for generating semantic ontology semi-automatically from Arabic text based on a lexical ontology model. They focused on the Arabic verbs. They used the Markov clustering algorithm to detect similarity between verbs and all the verbs synonyms. The basic corpus used is (معجم الغنى). The system morphologically analyzes the input verbs from the input corpus. They developed a tool called OntoArab maker. The evaluation was done using human expert [10].
Ahmed Cherif Mazari, et.al, [30] presented an approach to automatic ontology construction from a corpus of domain "Arabic linguistics". They reused extraction methods for extracting new terms that will provide elements of the ontology (concept, relation). Samhaa R. El-Beltagy, et.al, [31] introduced a model that automates the taxonomic agricultural construction process for domain ontology using a semi-structured domain specific web documents.

Although there are many explicit efforts to support and enrich the Arabic language, the characteristics of the Arabic language makes the work to support the language is not enough. The Arabic language is highly inflectional and derivational language. It includes huge numbers of vocabulary that not yet supported by the systems presented. The process of an automatic generation that deals with the language specialties still needs a lot of efforts. The time consumption of the manual ontology generation leads to the need for a way to accelerate the generation process with more accurate results. The rapid change in knowledge and science is one of the difficulties the manual generation faces. Arabic ontology systems need to be updated automatically to support the users with up-to-date information. The ontology learning (Automatic& semi-automatic) creation process helps to accelerate the generation of more accurate ontologies with the ability of easily updated systems.

The (Automatic & semi-automatic) Arabic ontology generation has supported the users but yet there are some gaps that need to be filled. Some of the systems focused on supporting a small part of the Arabic traditional grammar forgetting the ambiguity of the language and the need to support all the grammar branches. The ambiguity of the language the vowelization and the Polysemous need to be taking care of in the Arabic ontology systems generation. Also the process of the automatic evaluation the Arabic text needs a lot of efforts to reinforce the process of the evaluation and the systems evaluation.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3. NAAO ARCHITECTURE

NAAO architecture , as shown in figure 1, built from semi-structured data, that uses the same notations mentioned in [[7]] with modifications that suit the Arabic XML data sources. The architecture entails four main phases, **extrication, XML schema parsing, ontology generation and refinement and evaluation.** The following subsections discuss the framework in details.

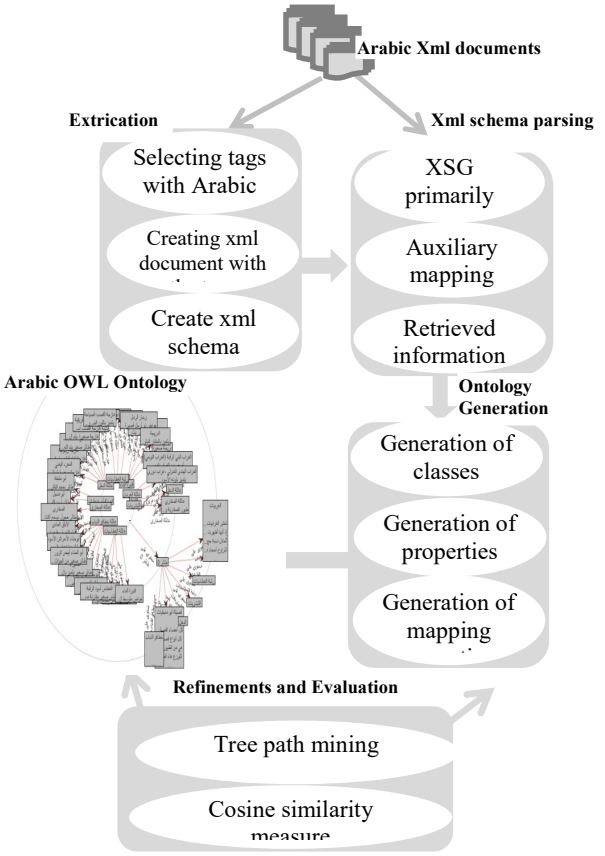

Figure 1: *Detailed Arabic Ontology FrameWork*

3.1. Extrication Phase

In this phase, all the information needed for the generation of the ontology is extricated from the input source. Arabic words are extricated from the XML files, associated with the connected tags. After that, the extracted tags are used to create XML schema. To build the required XML schema more than one XML documents are merged. Figure 2 shows the steps of the extrication phase.

Figure 2: *Steps of Extraction*

By going through the XML files, the Arabic words together with the associated tags are extricated. Then, the tags, extricated from every page, are linked to an XML document, and finally, an XML schema is created. The framework has the ability to merge more than one XML document in order to create the XML schema. The input is selected from three websites Arabic Wikipedia (https://ar.wikipedia.org), Saudi Wildlife (http://www.saudiwildlife.com/site/home/) and (http://www.uobabylon.edu.iq) websites in the domain of animals in Egypt and Saudi Arabia, In (10/1/2016).

3.2. XML Schema Parsing phase

It is the second phase of the ontology generation process that parses the input XML schema. The results of the parsing phase are XSG primarily version, auxiliary mapping, and retrieved information.

- **XSG primarily version**: it is a basic English language graph that is translated later into the Arabic language.
- **Auxiliary mapping**: it is derived from complex types that connect the classes with each other.
- **Retrieved information**: it is the information that is retrieved from the XML files to enrich the ontology classes.

Figure 3 and 4 show the steps of this phase. The input to the ontology is selected using similarity measurements.

If the input matches with 80% of these measures, the framework uses it, otherwise the input is eliminated from the ontology.

Journal of Theoretical and Applied Information Technology
15th January 2018. Vol.96. No 1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

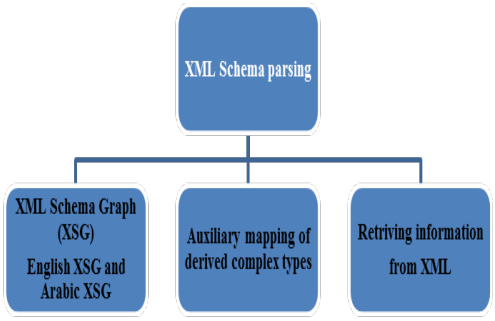

Figure 3: *XML Schema parsing*

Input: *XML schema graph XSG* **G= (N; E)**
generated from xml schema
Translate from ENGLISH to ARABIC
E is the edge set
For each N (element nodes, complex type nodes s and attribute nodes)
Connect each element e1 to its complex type CT
Connect each CT with its attributes and elements
End For
Auxiliary mapping derived type map
For each CT connect each derived complex type with its base (complex type) using (sub class of)

Figure 4: *XML Schema Parsing Algorithm*

3.3. Ontology Generation phase

The generated ontology is instance-free and contains the description of the class and its properties. In the generation phase, the OWL algorithm has been deployed for mapping procedures. These mapping procedures demonstrate how to change over every part of the XML schema to a semantically relating ontology segment. The Arabic ontology generation process involves the generation of classes, properties, and mapping connections.

The OWL classes are generated according to the algorithm in figure 5. This step is based on the Arabic XSG and the derived Map types. That is, the Arabic XSG is traversed to retrieve the vertices–for complex type nodes, element group vertices, and attribute group nodes. An OWL class is generated for each one of these nodes, then, the inheritance mechanism is used to set up subClass of relationships between classes.

The ontology generation process cannot be complete without mapping rules to illustrate how to convert the component of XML schema into its corresponding component in the ontology. On the first part of the ontology generation, OWL classes will be generated. There are three types of the OWL classes' global complex type, local named complex type, element group, and attribute group type. The OWL algorithm also includes the generation of the relations between subclasses. The second part of the ontology generation is the object properties; figure 6 shows the algorithm for the generation of object properties. The third part is the production of the data type properties.

Input: *XML schema graph XSG*
For each Global complex type
Map to: < OWL classes with the name of its type>
End For
For each Local named complex type
Map to: < OWL classes with the name of its surrounding element>
End For
For each Element groups and attribute groups
Map to: < OWL classes>
End For
For each inheritance mechanisms extension and restriction
Map to: complex type < subClassOf. > based complex type
End For

Figure 5*: OWL Classes generation algorithm*

3.3.1. Rules for object properties

Object properties emerge from subelement relationship. Figure 6 shows the algorithm for the ontology object properties. Element containment relationships in the schema will be translated into object properties in the Arabic ontology. If an element E1 contains another element E2 and the complex type of E1 and E2 are X1 and X2, then create object property. An object property is created such that its domain is the concept corresponding to X1 and its range is the concept corresponding to X2. The name of this object property is the concatenation of "has" with the name of range class. This rule does not affect the scope of the elements E1, E2 and the complex types X1, X2 within the schema. in other words, it does not matter whether any of the elements E1, E2 and the complex types X1, X2 are defined locally or globally.

For Element E containment relationships in the schema
Search for Element E with relationship R in XML schema
IF E is found **Translate** to object properties
IF an element E1 contains E2 && E1 € complex type X1 && E2€ complex type X2,
Then the object property domain € X1 && range € to X2.
End for

Figure 6: *Algorithm for object properties*

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3.3.2. Rules for Datatype Properties

Datatype properties emerge from attributes and simple types. Figure 7 shows the algorithm for the ontology datatype properties.

Element of simple type
Map to
Data type properties.
For each complex type CT contain an element E of a simple type ST (primitive or defined)
Create
DT data type property
If the simple type ST is a primitive XSD data type (xsd: string, xsd: integer, _ _ _)
Then
The domain is the class corresponding to the complex type CT.
The range of DP is this data type.
Else
ST is defined in the schema,
The domain is the class corresponding to the complex type CT.
The range of DP will be xsd: any Type.
End For
For each Elements of mixed complex type
If elements of this type contains a text node beside the sub-elements and/or attributes
Then Create
Data type property with
The domain the class corresponding to the surrounding complex type
The range the data type "xs: string"
End For

Figure 7: *Algorithm of Datatype properties*

3.3.3. Generation of Mapping connections

The ontology properties and the mapping connections are generated at the same time in this step. based on the XSG graph and the auxiliary mapping, and starting from the root node, the XSG is visited using depth-first. At each node, a particular treatment is performed according to the kind of the node being visited. Figure 8 shows the Arabic ontology connections rules.

1. **Class Connections:** Class connections are used to relate ontology class with XML nodes. Each class connection has a unique identifier within a mapping document.
   A class connection can be noted:
   Id = CC(C, Xpath)
   Where Id is the identifier of the connection, the symbol CC is used to indicate that this connection is a Class connection, the first argument C indicates an ontology term (in this case, class), and the second argument XPath indicates an XML node.

2. **Element (datatype properties) connections:** Element connections are used to relate elements with XML nodes. Each element connection has a unique identifier within a mapping document. The element connection belongs to exactly one class connection, called domain-concept-connection (DCC), which is the class connection of the element' domain. The element connection can relate element to one or more XML nodes, directly or via transformations, and, possibly, with conditions.
   A direct element connection is noted:
   Id = EC (E, Domain Class Connection, path)
   Where:
   Id is the identifier of the element connection,
   The symbol EC is used to indicate that this connection is an element connection. The first argument e is the mapped element. The second argument indicates the domain-class-connection.
   The third argument is the XPath expression of the mapped XML node.

3. **Object Property Connections:** An object property connection is used to relate two class connections to an object property. Each object property connection has a unique identifier within a mapping document. The two class connections that the object property connection relates are respectively called:
   - domain-class-connection, which is the class connection of the object property' domain, and
   - range-class-connection, which is the class connection of the object property' range.

   We say that object Property connection belongs to its domain-class-connection, and refers to its range-class-connection. Similar to other mapping connections, object property connections can have associated conditions.
   An object property connection is noted as Id = OPC (OP, Domain Class Connection-Id, Range Class Connection-Id)
   Where: Id is the identifier of the object property connection. The symbol OPC is used to indicate that this connection is an Object Property connection,
   The first argument OP is the mapped object property,
   The second argument Domain Class Connection -Id indicates the domain-class-connection.
   The third argument Range Class Connection -Id indicates the range-class-connection.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3.4. Refinement and Evaluation phase

Every step introduced may present wrong ideas or connections; along these lines, a refinement stage is required. Alternately, a refinement task can be presented at the end of each phase. Validation is frequently done by hand, but with the process of the ontology learning, it needs to be automated to accelerate the evaluation process. Ontology generation is not a static process; this process needs a lot of modification efforts to refine the ontology and deliver the required results—based on the data-driven evaluation. In this process, the ontology is evaluated by comparing it with the input XML documents. To achieve this, one could, for example, perform automated term extraction on the documents and simply count the number of terms that overlap between the ontology and the documents [11].

There are a variety of ways in which one could attempt to extract the information content of the documents to correlate that with the ontology. In our framework, we are going to use Term-based distance to measure that a document is commonly represented as a vector of terms in a vector space model (VSM). The contents of the vector space are compared to clear terms in a document gathering. Every vector corresponds in one document. The segments of the archive vector recalculate the weights of the comparing terms that correspond to their relative significance in the document and the entire document accumulation [11]. The mutual information between two terms *t*1 and *t*2 can be calculated by the *ontology-based VSM*. the *cosine similarity* to measure the term mutual information between their corresponding vectors is represented in equation 1:

$$\cos\left(\angle(t_1,t_2)\right)=\frac{t_1.t_2}{\|t_1\|\ \ |.|\|t_2\|}=\frac{\sum_{j=1}^{n}\tilde{x}_{j1}\tilde{x}_{j2}}{\sqrt{\sum_{j=1}^{n}\tilde{x}^2_{j1}}\ \sqrt{\sum_{j=1}^{n}\tilde{x}^2_{j2}}}$$

(Equation 1)

Where $\tilde{x}^2_{j1}$ and $\tilde{x}^2_{j2}$ represents the term weights of *t*1 and *t*2 in the document $\tilde{x}_j$ in the *ontology-based VSM*. According to the above cosine measure, the similarity of each pair of terms in the given document can be computed. [11]. the similarity measures between the input XML documents and the selected terms for the ontology generation are applied.

Detailed case studies for the insectivores and mammals animals have been discussed in previous researchers [1,12].

4. SYSTEM IMPLEMENTATION

The implementation of the framework is going to be discussed in this section. The framework has been implemented using Java programming language. JUNG (Java Universal Network/Graph Framework)is an open-source software library that provides a common and extendible language for the modeling, analysis, and visualization of ontology that can be represented as a graph or network.

4.1. Input Documents

Scientific classification of living organisms includes seven taxonomic groups: kingdom, phylum, class, order, Family, genus, and species. The kingdom is the largest taxonomic groups, while the smallest type is the species. For Every living organism a specific place known in the taxonomic group. The biggest kingdom is the animal kingdom that includes the largest number of elements. Vertebrates are animals with an internal backbone or spinal column. There are over 85,000 species of vertebrate animals such as amphibians, birds, fish, mammals, and reptiles. Only about 5% of all animal species are vertebrates. The remaining 95% of animals are invertebrates that don't have an internal backbone (e.g., insects, mollusks, and arthropods).

The input to the framework is the XML documents, and the output is our Arabic ontology. The input is selected from three websites Arabic Wikipedia (https://ar.wikipedia.org), Saudi Wildlife (http://www.saudiwildlife.com/site/home/) and (http://www.uobabylon.edu.iq) websites in the domain of animals in Egypt and Saudi Arabia. The input XML documents from these sites are all the animal kingdom types, the vertebrate and the invertebrate. The XML transformation to OWL ontology is a process of many steps. The model used 600 XML documents-that contain 6315 words. Table 1 indicates the animal types included in the Arabic ontology.

Journal of Theoretical and Applied Information Technology
15th January 2018. Vol.96. No 1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 1: *Animal XML documents analysis before extraction*

	Ontologies	# docs	# elems
The vertebrate's ontologies (الفقاريات)	Mammal's (الثدييات)	196	2105
	Bird's (الطيور)	142	1445
	Reptiles (الزواحف)	54	549
	Amphibians (البرمائيات)	31	422
	Fish (الاسماك)	154	1587
	rate's (اللافقاريات)	23	207

4.2. Ontology generation

In order to build the ontology, the framework sending initial keywords to search for it on the three websites and retrieves the related pages. The returned pages are parsed to determine the useful text from each site and try to find the initial keywords in it. After parsing the XML, the framework generates the XML schema and the XML schema graph for these pages. From this chart, the framework can determine the classes, the parent-child relations, and the elements. The framework also uses Frequency-based mining [11] to separate important labels concepts (class) from elements. It mines parent-child relationships from semantically partitioned Web pages. This mining is done using frequent tree mining algorithms to find the (isa) relationships among the concepts.

4.3. NAAO Output and analysis

When the word animal is sent as an initial keyword, the framework selected the related XML documents and parsed them. After parsing the corresponding pages, the system generates the schema and the XML schema graph. It contains an illustration of classes, the relation between classes, the subclass of relations and the connection between the classes, elements, and properties. Then it uses frequency-based mining to make sure the generated graph is in the right form. Finally, it connects the documents and removes the duplication between the classes and elements. The framework used 600 XML documents ,after removing the duplicating, 1916 concepts, 4439 elements and 3469 relationships is generated from the first parsing, then after applying the frequency based mining, the ontology contained 1803 concepts, 4016 elements, and 3212 relationships.

The generated ontology is evaluated using equation no.1 it reflects the similarity between the input XML and the output ontology using the cosine similarity measure. Figure 8 and Table 2 shows the results of the generated ontology.

The output has divided the Arabic ontology into vertebrate and invertebrate animals. The vertebrate animals include five ontology cases, and the invertebrate animals include one ontology case. Both vertebrate and invertebrate are represented in graphical form. Figure 11 shows part of the generated graphical representation of the bird's ontology, the figure includes the part of the concepts, elements, and relations.

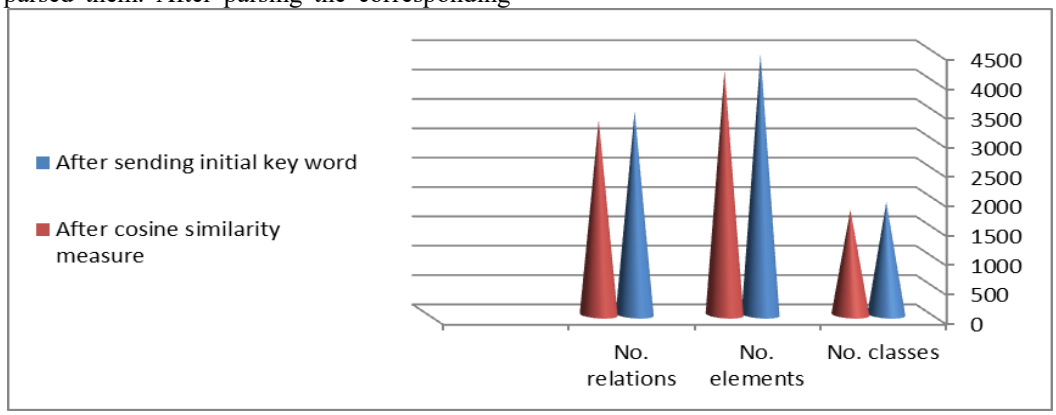

Figure 8: *Arabic Ontology output analysis*

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 2: Arabic Ontologies output analysis

	Ontologies	No. classes	No. elements	No. relations
The Vertebrate's ontologies output (الفقاريات)	Mammal's (الثدييات)	540	1520	1070
	Bird's (الطيور)	480	1270	790
	Reptiles (الزواحف)	311	456	368
	Amphibians (البرمائيات)	25	77	133
	Fish (الاسماك)	220	513	325
	The invertebrate's (اللافقاريات)	320	603	783

4.3.1. The vertebrate animals

The vertebrate ontologies; the input to the vertebrate ontology is selected from two websites Arabic Wikipedia (https://ar.wikipedia.org) and Saudi Wildlife (http://www.saudiwildlife.com/site/home/) websites in the domain of animals. **For the mammal's** ontology, it contains 255 XML input documents, resulting 540 classes and 1520 element, 1070 relation between classes and subclasses. The connection between the classes, elements, and properties are illustrated in the generated ontology. The cosine similarity measure for the mammal's ontology is 395 concepts, 1452 element, and 1009 relationship. **The bird's ontology, it** contains 250 XML input document, resulting 480 classes and 1270 element the 790 relations between classes the subclass, the connection between the classes, elements and properties are illustrated in the generated ontology. The cosine similarity measure for the mammal's ontology results in 431 concepts, 1205 element, and 784 relationships. **The reptile's ontology,** it contains 48 XML input document, resulting 311 classes and 456 elements, 368 relations between classes and subclass~~,~~ The connection between the classes, elements, and properties is illustrated in the generated ontology. The cosine similarity measure for the mammal's ontology results ~~is~~ 301 concepts, 401 elements, and 377 relationships. **The amphibian's ontology**, it contains four XML input document, resulting 25 classes and 77 elements the 133 relations between classes the subclass, the connection between the classes, elements, and properties are illustrated in the generated ontology. The cosine similarity measure for the mammal's ontology results is 22 concepts, 69 elements and 134. **The fish ontology,** it contains 52 XML input document, resulting 220 classes and 513 elements the 325 relations between classes the subclass, the connection between the classes, elements, and properties are illustrated in the generated ontology. The cosine similarity measure for the mammal's ontology results is 197 concepts, 456 elements and 267. Figure 9 represents part of the graphical representation for the generated Arabic ontology.

4.3.2. The invertebrate animals

The invertebrate's ontology; the input to invertebrates ontology is selected from (http://www.uobabylon.edu.iq) and Arabic Wikipedia (https://ar.wikipedia.org) websites in the domain of animals. It contains 103 XML input document resulting 320 classes and 603 elements. The 783 relations between classes and subclass, the connection between the classes, elements, and properties are illustrated in the generated ontology. The cosine similarity measure for the invertebrate's ontology results is 311 concepts, 570 elements, and 746 relationships.

With this evaluation results for the invertebrate and vertebrate animals ontologies, the proposed Framework found to be efficient in extracting Arabic terms and generating Arabic ontology. The framework differs from previous systems with the fully automated creation, the coverage of animal ontology terms and the unique creation methods. using 600 XML documents ,after removing the duplicating, 1916 concepts, 4439 elements, and 3469 relations is generated from the first parsing, then after applying the frequency based mining, the ontology contained 1803 concepts, 4016 elements, and 3212 relations. These results can be increased in the future work to include more relationships and concepts.

Journal of Theoretical and Applied Information Technology
15th January 2018. Vol.96. No 1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

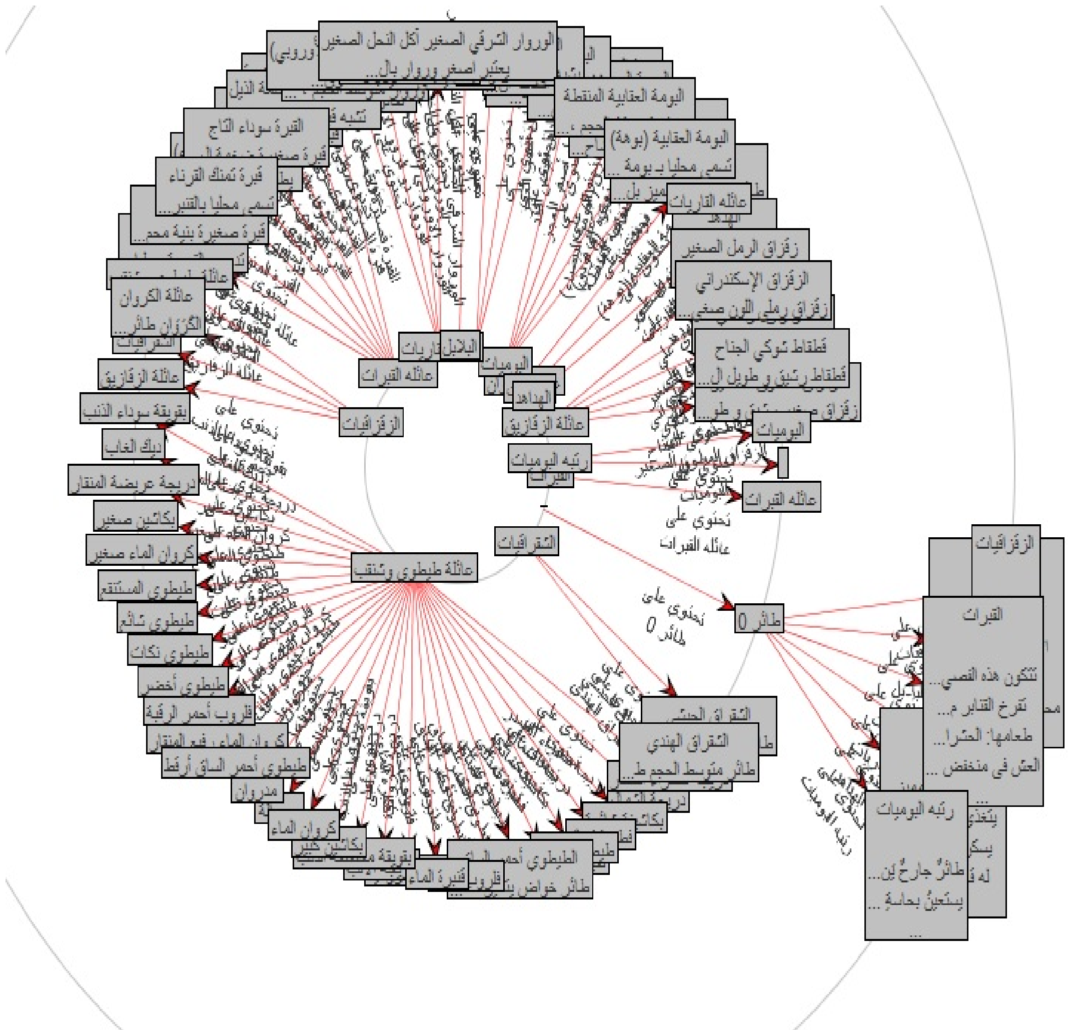

Figure 9: *Graphical representation of part of the generated Arabic ontology*

5. CONCLUSION

Due to the progression of the Arabic web contents and Arabic users, it is vital to manipulate such data in an efficient way to retrieve effectively the desired results. Arabic ontology generation is a very important field which helps in the construction and retrieval of Arabic information. There are various studies conducted in the Arabic language in Semantic Web. But the support of the vast language vocabulary yet is not enough–and the Automation process is maintained by only a few systems. Although, Arabic ontology generation aids in multiple domains by constructing and retrieving pertinent information more precisely to the user requirements yet the support in the Arabic language is very weak. This paper introduced an Automatic Arabic ontology generation framework (NAAO) that showed by empirical analysis to automate the creation process of Arabic ontology. Such automation process accelerates the Arabic ontology generation, (using semi-structured data XML files).NAAO consists of four main phases (extrication, XML schema parsing, ontology generation and refinement and evaluation). Extrication phase, where all the data required for the ontology generation is extricated from the information source. XML schema parsing phase, where the extracted data is going to be investigated and the hierarchical analysis to identify all the parts of the ontology is going to be created. Ontology generation phase deals with the production of the Arabic ontology (classes, elements, and relations). Finally, refinement and evaluation phase that deals with improvements of the Arabic ontology.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

NAAO enjoys certain advantages than others. It provides mapping connections to ensure the right relationships with the classes and the elements to its corresponding nodes in the XML documents. It offers an automatic evaluation of the generated Arabic ontology using cosine similarity measurements. NAAO has been tested with 600 XML file–for ontology generation based animal kingdom as a case study. The ontology has been divided into two parts vertebrates and invertebrates. The vertebrate's ontology includes 5 smaller ontologies mammals, birds, fish, reptiles, and amphibians. It contains 1896 classes (concepts), 4439 elements and 3469 relationships. The invertebrates ontology results contain 311 concepts, 570 elements, and 746 relationships. The paper provides the empirical analysis of the NAAO ontology and in the future work, we are going to support analytical analysis of the framework.

REFERENCES:

[1] Dalia Fadl, Safia Abbas, and Mostafa Aref, "Approach for Automatic Arabic Ontology Generation", International Journal of An Intelligent Computing and Information Sciences (IJCICIS) Faculty of Computer and Information Sciences, Ain Shams University, Cairo, Egypt,2016.

[2] L. Al-Safadi, M. Al-Badrani, M. Al-Junidey, "Developing Ontology for Arabic Blogs Retrieval," International Journal of Computer Applications (0975 – 8887) Volume 19– No.4, April 2011.

[3] Annika O¨ hgren, "Ontology Development and Evolution: Selected Approaches for Small-Scale Application Contexts", Information Engineering Group Department of Computer and Electrical Engineering School of Engineering, J¨onk¨oping University .J¨onk¨oping, SWEDEN, ISSN 1404-0018, 2005.

[4] Maryam Hazman, Samhaa R. El-Beltagy, Ahmed Rafea, "A Survey of Ontology Learning Approaches", International Journal of Computer Applications (0975 – 8887) Volume 22– No.9, May 2011.

[5] Maryam Hazman, Samhaa R. El-Beltagy, Ahmed Rafea "Ontology Learning from Textual Web Documents", INFOS2008, March 27-29, 2008 Cairo-Egypt.

[6] N. Ghneim, W. Safi, M. Al Said Ali," Building a Framework for Arabic Ontology Learning", Damascus University, Damascus, Syria, 2008.

[7] R. Ghawi, and N. Cullot, "Building Ontologies from XML Data Sources", In 1st International Workshop on Modelling and Visualization of XML and Semantic Web Data (MoViX '09), held in conjunction with DEXA'09 (Linz, Austria, September 2009).

[8] Sojka, Choi, Fellbaum and Vossen eds, "Introducing the Arabic WordNet Project", in Proceedings of the Third International WordNet Conference, 2006.

[9] Hegazy, M. Sakre, Eman Khater," Arabic Ontology Model for Financial ", Procedia Computer Science, Volume 62, 2015, Pages 513-520.

[10] Bedr-eddine Benaissa, Djelloul Bouchiha, Amine Zouaoui, Noureddine Doumi, "Building Ontology from Texts", Procedia Computer Science, Volume 73, 2015, Pages 7-15

[11] H. Hjelm, "Cross-language Ontology Learning", Printed in Sweden by US-AB, Stockholm ISBN 978-91-7155-806-0, 2009.

[12] Dalia Fadl, Safia Abbas, and Mostafa Aref, Automatic Arabic Ontology Construction framework: Insectivore's case study, AISI 2017: Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2017 pp 458-466, Cairo, Egypt.

[13] C.e Roche, "ONTOLOGY: ASURVEY", University of Savoie Equipe Condillac - Campus Scientifique, 73 376 Le Bourget du Lac cedex – France, 2002.

[14] N. Noy and C. Hafner, "The State of the Art in Ontology Design", AI Magazine Volume 18 Number 3,1997.

[15] H. Aliane, Z.Alimazighi, Mazari A.Cherif, "Al –Khalil: The Arabic Linguistic Ontology Project", Semantic web and Arabic Language Team, Research Center on Scientific and Technical Information, Algiers. 2010.

[16] N. Ghneim, W. Safi, M. Al Said Ali, "Building a Framework for Arabic Ontology Learning", Damascus University, Damascus, Syria, 2008.

[17] T. R. Gruber, "Toward Principles for the Design of Ontologies", Stanford Knowledge Systems Laboratory, 1996.

[18] Haytham T. Al-Feel, Magdy Koutb, and Hoda Suoror, "Semantic Web on Scope: A New Architectural Model for the Semantic Web, Journal of Computer Science 4(7): 613-624, 2008.

[19] Tim Berners-Lee, James Hendler and Ora Lassila,"The Semantic Web", Scientific American, May 17, 2001

[20] T. R. Gruber, "A Translation Approach to a Portable Ontology Specification", Knowledge Acquisition, vol. 6, pp. 199-221, 1993.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[21] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, andK. J. Miller" Introduction to WordNet: an on-line lexical database",International Journal of Lexicography3(4):235-244(1990).
[22] Aya M. AlZogby, Ahmed Sharaf Eldin Ahmed, Taher T. Hamza, "Arabic Semantic Web Applications –A Survey ", Semantic Web Journal, 2012.
[23] P.Saariluoma, K. Nevala, “From Concepts to Design Ontologies”, Cognitive Science, University of Jyväskylä, Finland, 2009
[24] L. Al-Safadi, M. Al-Badran, M. Al-Junidey, “Developing Ontology for Arabic Blogs Retrieval”, International Journal of Computer Applications (0975 – 8887)Volume 19– No.4, April 2011
[25] Aya M. AlZogby's, Ahmed Sharaf Eldin Ahmed and Taher T. Hamza “Arabic Semantic Web Applications – A Survey”, 2013 Journal of Emerging Technologies in Web Intelligence, Vol 5, No 1 (2013), 52-69, Feb 2013
[26] Hend S. Al-Khalifa, Areej S. Al-Wabil, “The Arabic language and the semantic web: Challenges and Opportunities”, The 1st International Symposium on Computers and Arabic Language & Exhibition 2007.
[27] A. De Nicola, M. Missikoff, and R. Navigli, “A Software Engineering Approach to Ontology Building”, Information Systems, 34(2009), pp. 258–275.
[28] Mustafa Jarrar, "Building a Formal Arabic Ontology", In Proceedings of the Experts Meeting on Arabic Ontologies and Semantic Networks April 26-28, 2011.
[29] Ahmed Cherif Mazari, Hasina Alliance, and Zaia Alimazighi,"Automatic construction of ontology from Arabic texts", Proceedings ICWIT,2012.
[30] Samhaa R. El-Beltagy, Maryam Hamza, Ahmed Rafea, "Ontology learning from domain specific web documents", vol. 4, No.1/2, May 2009.
[31] Hassina Aliane, Zaia Alimazighi, Mazari Ahmed Cherif,"Al–Khalil: The Arabic Linguistic Ontology Project",Seventh International Conference on Language Resources and Evaluation, 2010.