Journal of Theoretical and Applied Information Technology
15th November 2018. Vol.96. No 21
© 2005 – ongoing JATIT & LLS

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

ARABIC ONTOLOGY-BASED APPROACH FOR CHEST DISEASES DIAGNOSIS

[1]ALI ALNADER, [2]ISSAM SALMAN, [3]KHALIL AJAMI, [4]AMMAR ALZEIN

[1,3]Syrian Virtual University, Web Science program, Syria

[2]Czech Technical University, Department of software engineering, Czech

[4]Damascus University, Department of Internal Medicine, Syria

E-mail: [1]ali_70619@svuonline.org, [2]issam.salman@fjfi.cvut.cz, [3]t_kajami@svuonline.org, [4]a.w.zein@gmail.com

ABSTRACT

Chest diseases are a subgroup of respiratory system diseases. The symptoms of these diseases are similar and this makes the diagnosis process difficult. Therefore, to ease the diagnosis process we gathered Information about fourteen diseases, which attack the chest, with their symptoms and investigations about them. In this paper, we present an Arabic ontology-based approach for chest diseases diagnosis. We focus on ontology building process. This ontology can be used to help physicians and other users, determine the chest disease that a patient is suffering from and what are the investigations that should be applied. While experts can easily gather information from this data, lay users lack the expertise needed to deal with it. Most of the efforts to solve this problem focus on the English language. So also, in this paper, we present a new approach based on natural language processing to translate Arabic language query to SPARQL query

Keywords: *Knowledge Representation, Semantic Web, Ontology Development, Arabic Language, SPARQL, Natural Language Processing*

1. INTRODUCTION

No one can ignore the importance of the respiratory system as one of the most important systems in the human body, which provides it with oxygen. Therefore, any trouble in its function will lead to death.

Several diseases can infect the respiratory system, each of them attacks at least one organ and has a set of signs and symptoms. Therefore, the early detection of any defect in the respiratory system functions stands at the top of doctor's tasks.

However, the diagnostic process is not an easy process due to the complexity of the human body and overlapping phenotypes. For that reason, computer science can be helpful to support physicians in the diagnosing's process. So, creating a system based on ontology will facilitate the diagnosis for doctors, by providing it with the patient's symptoms and signs, it will query the ontology and provides feedback about the expected disease. Since there is no research- to our knowledge- that uses ontologies to serve chest diseases in Arabic.

In addition, we propose a natural language processing model, which will enable users to query the semantic data using their natural language. Our approach starts a translation process by removing stop words then identifying the query target after that form triples by mapping the rest of words in the query or their stems or their meanings with the ontology.

The rest of this paper gives a background about ontology development, disease diagnosis and translating natural language queries to SPARQL queries, presents the methodology of ontology development, the implementation of the ontology using Protégé tool and presents the proposed model, and the evaluation of the ontology. Finally, contains the conclusion and future work.

2. CHALLENGES AND IMPORTANCE OF THE RESEARCH

The challenge in designing such an ontology is that it requires medical experts to revise and evaluate the gathered information. and the challenges in the field of natural language processing of Arabic language is the complexity from syntactical level including the convoluted Arabic grammar.

There are different word derivations and there is an impact of the diacritics marks on the word's meaning. Besides, there are other challenges related to the usage of the stem, in which a word's stem provides words with different meanings. The contributions of this research are represented in:

- Design Arabic ontology for chest disease diagnosis.
- Help lay users to query the ontology using their natural language.
- Enrich Arabic content on the web since it is an Arabic ontology.

3. RELATED WORKS

In this section we discus many works in the domain of ontology development, disease diagnosis and translate natural language query to SPARQL query and vice versa.

3.1. Ontology development

In [1] researchers suggested a methodology consists of three basic modules: the diagnostic module, the staging module and the treatment recommendation module. In order to detect patient disease, the patient provides his/her signs and symptoms to the diagnostic module, which detects what type of cancer the patient is suffering from. Then the staging module finds the current stage of cancer based on the cancer type, signs and symptoms that are provided by the patient. Based on the determined cancer type and cancer stage, the treatment recommendation module can recommend a specific treatment for the case.

In [2] researchers discussed the problem of knowledge acquisition. They proposed an inference method according to the case at hand. The system takes the ontology and the symptoms as input. Then in the first phase, the system returns a set of diseases. Thus, if the number of these returned diseases is one, then the process of diagnosing is finished. However, if the number of returning diseases is greater than one, the differential diagnosing process begins where the system retrieves similar cases through semantic way depending on a knowledge model.

In [3] Cristina Romero Tris proposed a decision support system based on a knowledge base to help physicians in the diagnosis process, and to verify the diagnosis made by the doctor. Therefore, if the doctor inputs a disease not related to the symptoms, the system will notify the doctor of the inconsistency. Moreover, it will suggest the disease that best fits the symptoms. The proposed system aims to personalize existing knowledge base through the extraction of a partial ontology, which contains the medical terms that refer to the patient.

In [4] Lakshman Jayaratne proposed a decision support system based on ontology. The proposed system composed of two components, a genotype component and phenotype component. The initial input of the system is genetic sequences of a patient. Firstly, the system identifies whether the gene in input is mutated or not with respect to the referenced gene sequences. Then, according to those mutations, a corresponding common list of phenotypes are identified and shown to the physician. They are listed down according to the frequency and probability of phenotypes that might occur. The physician must first make sure that the phenotypes given by the system are really in the patient’s body and can then get the diagnostics report after choosing the matching phenotype from those listed by the system.

In [5] Researchers proposed a framework composed of three phases wherein the first phase is a text analysis technique would be applied to medical records. In the second phase, the semantic analysis process was applied to store the information extracted from the first phase in a knowledge base, and the last phase involves querying the knowledge base to get the related medical information. Here, they used medical rules to infer additional medical facts about the patients and to generate a rich knowledge base of patient facts.

In [6] Researchers proposed an ontology-based approach to diagnose the date palm disease and suggest appropriate treatment by identifying anomalous observations on the parts of the tree. The approach consists of three interrelated modules: knowledge base, reasoning engine, and a server-side application. The knowledge base was built using OWL ontology and contains information related to date palm. The reasoning engine accepts user input queries and responses to data through the I/O interface by analyzing the acquired dynamic information together with the static knowledge stored in the knowledge base. In addition, the ontology was evaluated using the task-based framework and it indicates an accuracy of 100% and 97.6% when using the precision and recall method.

3.2. Translating Natural Language Query to SPARQL Query

In [7] researchers proposed a generic approach to translate Arabic NL queries to SPARQL. The proposed approach utilizes the off-the-shelf Arabic language toolkit to build the parse tree of the user query. It then analyses the syntactic structure of the tree to extract NPs, identify head and modifier words and represent the query in a triple format, i.e. subject-predicate-object. The proposed approach is independent from the ontology since it can be easily ported from one ontology to another.

In [8] researchers proposed a new version of LODQA which takes as an input natural language query and returns it SPARQL form. It consists of three modules, which are:

- First Module is responsible for handling the natural language query, parsing it, and producing a graph representation called as pseudo graph pattern (PGP), which contains nodes and relations. Typically, the nodes correspond to the nouns and relations to the dependency paths between them. Additionally, PGP specifies the focus of the query.
- Second Module is responsible for finding the URIs and values of the nodes in the PGP through mapping PGP with the ontology. A natural language term may be normalized to more than one RDF terms due to ambiguity. Therefore, more than one Anchor PGPs may be produced.
- Third Module is responsible for searching the target dataset for corresponding parts, considering possible variations. To absorb structural discrepancy between the APGP and actual structure in the target dataset, it attempts to generate SPARQL queries for all the possible structural variations. Then the SPARQL queries were submitted to the target endpoint and the answers were gathered and provided to the user.

In [9] researchers proposed an approach to translate English queries to SPARQL queries based on knowledge stored in ontologies and lexicons. The translation process composed of seven steps, which are:

- Delete stop words
- Stem the remaining words in the query and delete punctuation and extract dates.
- Map the remaining tokens against the ontology. Concepts, individuals, relations and properties are matched, the appliance labels are extracted.
- Tokens without corresponding entries in the ontology are checked against the lexicon. These are either KPIs or keywords, which do not belong to the ontology.
- After identifying the tokens in the ontology or lexicon, the corresponding SPIN rule, which provide the main structure of the SPARQL query, is extracted.
- The last step merges the defined SPARQL body from the SPIN rule with the additional information from the ontology and the extracted dates.

In [10] researchers proposed SPARQL2NL, a novel approach that can verbalize SPARQL queries. The approach is tailored towards SPARQL constructs typically used in keyword search and question answering. It consists of four main steps: a preprocessing step which normalizes the query and extracts type information for the occurring variables. Processing step which constructs a generic representation of the query. A post-processing step, which applies reduction and replacement rules in order to improve the legibility of the verbalization, and a realization step, which generates the final natural language representation of the query.

In [11] researchers proposed an approach that enabled end-users to query knowledge bases. This approach is based primarily on predefined samples that lead user query translation. The researchers relied on the hypothesis that end-user queries varied among fixed queries. The process of translating the natural language query into the SPARQL query proposed in this search consists of two basic steps. The first step was converting the natural language query to Pivot Query, Where the nominal entities were selected, the dependencies were analyzed, the target was derived from the query, and a set of predefined rules was applied to the reliability statement to extract all elements of the query. The second step involved mapping the predefined samples with the Pivot query. A set of possible interpretations of the user query, which were arranged according to their expected relationship with the user query, were then suggested to the user as a set of rewritten queries.

In [12] researchers proposed a prototype version of LODQA (Linked Open Data Question Answering), which build SPARQL query through two steps. In the first one, it uses Enju English parser, which produce predicate-argument relation, and in the second step, LODQA searches the ontology for corresponding terms. Researchers used OntoFinder, which search in medical ontologies.

4. ONTOLOGY DEVELOPMENT

In this section, we present the steps which we follow to build our ontology. The proposed CDDOnto (Chest Disease Diagnostic Ontology) will be very important for diagnosing chest diseases. We gather the ontology content from [13] and from a domain expert who helps to identify concepts and relationships between them. There is a wide range of tools available for creating ontologies such as SWOOP, Onto Track and protégé, but we choose protégé because according to [14] it is the most domain-independent tool. According to [15], building an ontology requires executing the following steps:

- Determine the domain of the ontology

We cannot build an ontology without specifying its domain. The domain of our ontology is diagnosing chest diseases.
Our ontology can answer different types of questions like:

- o What are the symptoms of a given disease?
- o What are the investigations of a given disease?
- o What is the disease of giving symptoms?

- Reuse Existing Ontologies

We build CDDOnto from scratch since there is no such ontology in the Arabic language.

- Overview the Ontology

We gathered information about fourteen diseases, their symptoms and investigation. Then we organized them in an ontology. Figure 1 shows all classes of the ontology.

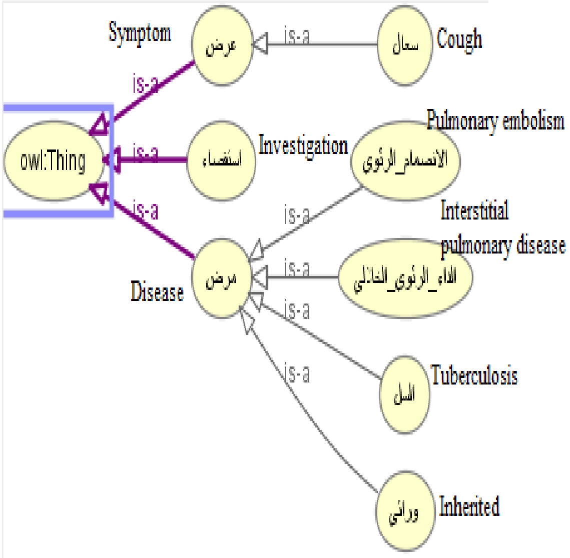


Figure 1 Ontology Classes.

- Enumerate main terms in the ontology

We find three main classes, a disease (مرض) which has four subclasses, symptom (عرض) which has one subclass and investigation (استقصاء).

Table I: Ontology Classes

Name in English	Name in Arabic
Investigation	استقصاء
Symptom	عرض
Disease	مرض
Cough	سعال
Pulmonary embolism	الانصمام الرئوي
Interstitial pulmonary disease	الداء الرئوي الخلالي
Tuberculosis	السل
Inherited	وراثي

- Define the Properties of the Classes

There are two types of properties: object property, and data property. While the first links object to object, the other link object to XML schema. We add data properties such as اسم, توصيف, سبب to the ontology. They are used for giving a value to an instance of a class. For example, "سبب" cause of "االتدرن البدئي" primary tuberculosis is Mycobacterium tuberculosis "المتفطرات السلّيّة". The data property is applicable to each instance of the class. Table 1 illustrates the object properties, while table illustrates data properties.

Table 2: Object Properties of the Classes

Object Property	English Name	Domain	Range
له_عرض	has_symptom	مرض	عرض
عرض_لـ	symptom_of	عرض	مرض
يُشخِص	Diagnosis	استقصاء	مرض
يشخص_بـِ	diagnosed_by	مرض	استقصاء

Table 3: Data Properties of the Classes

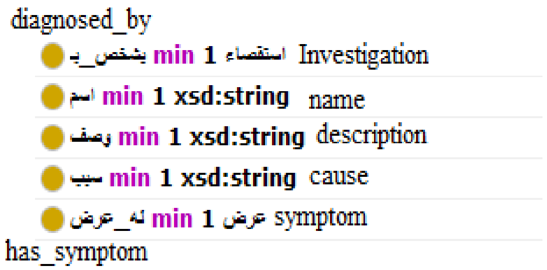
Data property	In English	Domain	Range
اسم	Name	مرض، عرض، استقصاء	String
سبب	Cause	مرض	String
توصيف	Description	مرض	String

- Define the attributes of the data properties

Each data property has attributes and each of them is specifying different things such as value types, allowed types and the cardinality of the values, which attribute can take.

- o Value type: describes the types of values a property can have. For example, property "سبب" has a value of type string.

- o Allowed values: describes the allowed values for the properties. For example, the allowed values for the property "*اسم*" are "مرض","عرض" and "استقصاء".
- o Cardinality: defines how many values a property can have. For example, the property "اسم" has multiple cardinalities, but it has at least one. Figure 2 shows the value type and cardinality of some of the properties.



diagnosed_by
● يشخص_بـ min 1 استقصاء Investigation
● اسم min 1 xsd:string name
● وصف min 1 xsd:string description
● سبب min 1 xsd:string cause
● له_عرض min 1 عرض symptom
has_symptom

Figure 2: Data Cardinality of Data Properties.

- Creating Instances of CDDOnto

The creation of individuals allows all the properties of the classes to be recorded. We defined 101 instances that represent all ontology concepts, including diseases, symptoms, and investigations.

- Applying Ontology Reasoner

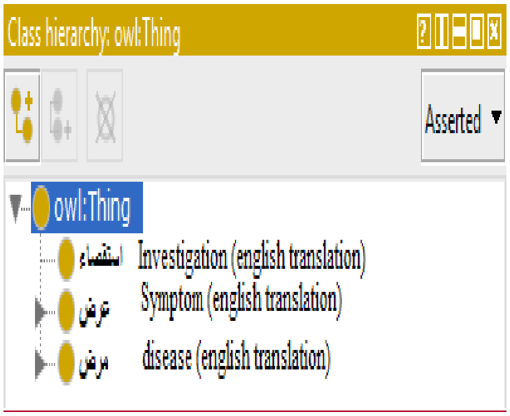
We apply an ontology reasoner (e.g. Hermit reasoned) to check that everything is ok and to identify new relations from existing ones.

5. CDDONTO IMPLEMENTATION IN PROTÉGÉ

This section describes the development of CDDOnto in protégé as an owl ontology.

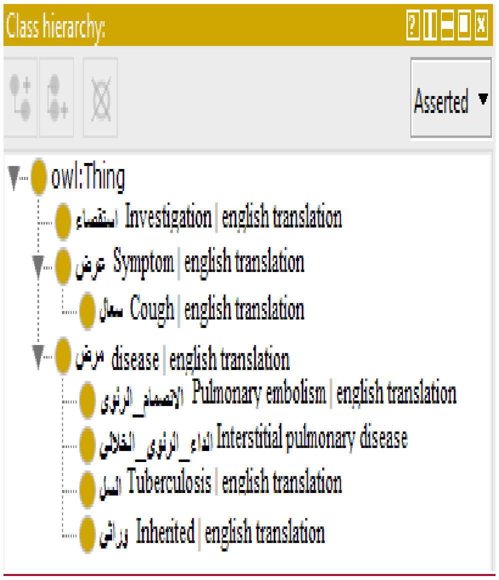
5.1. Classes and subclasses

In CDDOnto disease (مرض), symptom (عرض) and investigation (*استقصاء*) are subclasses of class Thing. Figure 3 shows the main classes in the ontology, whereas figure 4 shows all.



owl:Thing
استقصاء Investigation (english translation)
عرض Symptom (english translation)
مرض disease (english translation)

Figure 3: Protégé snapshot of the class hierarchy.



owl:Thing
استقصاء Investigation | english translation
عرض Symptom | english translation
سعال Cough | english translation
مرض disease | english translation
الانصمام_الرئوي Pulmonary embolism | english translation
الداء_الرئوي_الخلالي Interstitial pulmonary disease
السل Tuberculosis | english translation
وراثي Inherited | english translation

Figure 4: Taxonomy of main classes.

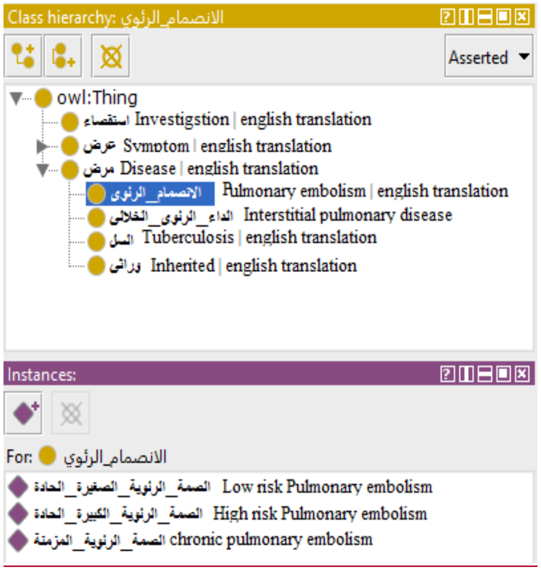
5.2. Instances, Properties, and Facets

In CDDOnto, individuals are defined with their data properties. In addition to object properties between them. Figure 5 shows data taxonomy such as "الانصمام الرئوي" pulmonary embolism class, which contains three instances. Pulmonary embolism is a blockage in the pulmonary artery, which supplies the blood to the lungs. Data Properties are shown in Figure 6, which contains three properties, these properties, are explained as follows:

اسم “name”: the اسم refers to the word of circulation of the disease in the world.

- o سبب "Cause": express an event in the human body, which Cause a disease.

- وصف "Description": describes either a disease or an investigation.



Class hierarchy: الانصمام_الرئوي

Asserted

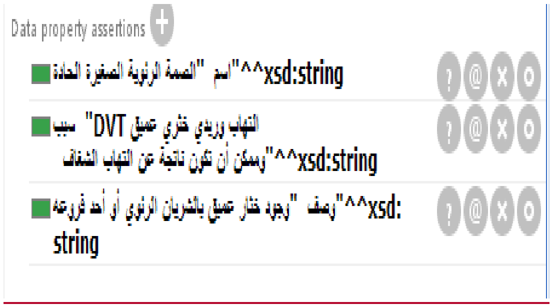
- owl:Thing
  - استقصاء Investigstion | english translation
  - عرض Svmptom | english translation
  - مرض Disease | english translation
    - الانصمام_الرئوى Pulmonary embolism | english translation
    - الداء_الرئوى_الخلالى Interstitial pulmonary disease
    - السل Tuberculosis | english translation
    - وراثى Inherited | english translation

Instances:

For: الانصمام_الرئوي

- الصمة_الرئوية_الصغيرة_الحادة Low risk Pulmonary embolism
- الصمة_الرئوية_الكبيرة_الحادة High risk Pulmonary embolism
- الصمة_الرئوية_المزمنة chronic pulmonary embolism

Figure 5: Instances of pulmonary embolism.



Data property assertions

- 🟩 اسم "الصمة الرئوية الصغيرة الحادة"^^xsd:string
- 🟩 سبب "DVT التهاب وريدي خثري عميق يمكن أن تكون ناتجة عن التهاب الشغاف"^^xsd:string
- 🟩 وصف "يوجد خثار عميق بالشريان الرئوي أو أحد فروعه"^^xsd:string

Figure 6: Data properties of an individual.

6. ARABIC TO SPARQL TRANSLATION APPROACH

In this section, we will go through the steps of the proposed approach, which is illustrated in the figure 7.

6.1. Removing stop words

The stop words were known from the beginning of information retrieval, which are the most commonly used words in the language and are not meaningful but are used to complete it, such as subjective pronouns, demonstrative pronouns, prepositions, negation, WH questions, adverbs and adverbs of place. Deleting these words will not affect the content of the query.

6.2. Identifying query goal

We mean by the query goal the word, which come after the question word, and it, will be the first word after the previous step (remove stop word). It is mapped to one of the ontology classes or data properties. After that, the goal is passed to light stemming process like removing the "الـ" from the beginning of each word.

6.3. Generating NGram

NGram is a sequence of N elements of a given text string, used in several areas, such as determining the probability of automatic machine translation, predicting the next most likely word in a sentence, checking the spelling of text, or identifying the feeling of part of a given text [16].

6.4. Getting Ontology concepts

In this step, we query the ontology to get all ontology classes and data properties, and then we normalize them by removing the underscore (_) between the terms. After that, we stem them all using the same stemming algorithm which is used in stemming the query terms to get shared roots between the query terms and ontology concepts.

6.5. Mapping Query terms to Ontology concepts

In this step, we map the query terms, one of their meanings using Arabic WordNet, their stems or one of the meaning of their stems to the ontology concepts, and when we find a correspondence, we replace the query term with it. The stemming process is useful for obtaining the same root for different word forms. There are different algorithms available for Arabic word stemming but, in this research, we use ISRI Arabic stemmer [17].

6.6. Specifying query Individuals

In this step, we specify all query individuals resulted from the previous step. We can specify Individuals in the ontology through the query, which is illustrated in the figure 8.

6.7. Generating Duals

In this step, we group the query goal with each individual to form pairs (query goal, Individual).

6.8. Generating Triples

In this step, we generate the triples using the duals generated in the previous step. We get the relation linked between them through querying the ontology. It is known that each relation has two sides, domain and range, and we can get the property using the query, which is shown in Figure 9. Note that the line 2 in the figure 8 and 9 is the ontology URI.

In case the range of the natural language query is an individual, the previous query will return an empty result set. Therefore, we use the query shown in the Figure 10 to get the type of the individual then use the query that is shown in Figure 9 to get the relation.

6.9. Forming the Query

The SPARQL query consists of three sections, which are: SELECT statement, WHERE clause and solution modifiers. In this research, we focus on forming the first two sections and we do not discuss the forming of the solution modifier section. The RDF triple that was built in the previous step forms the WHERE clause for the SPARQL query. Now we have to construct the SELECT statement and associate it with WHERE clause to form a meaningful SPARQL query. To construct the SELECT statement, we have to identify the target of the query which we specified in step 2, now we have to link the target of the query with the WHERE clause. We have done this by the following:

- Add a projection variable to the SELECT clause which we name it as ?goal.

- The forming of the WHERE clause differs depending on the purpose of the query. In case the purpose of the query is a data property then the WHERE clause is formed as following (:Individual, :data_property, goal). Then we replace all occurrences of the goal with the variable ?goal. After that, we add the Ontology URI to all expressions in the query - except the variables- and all required namespaces as prefix to the query.

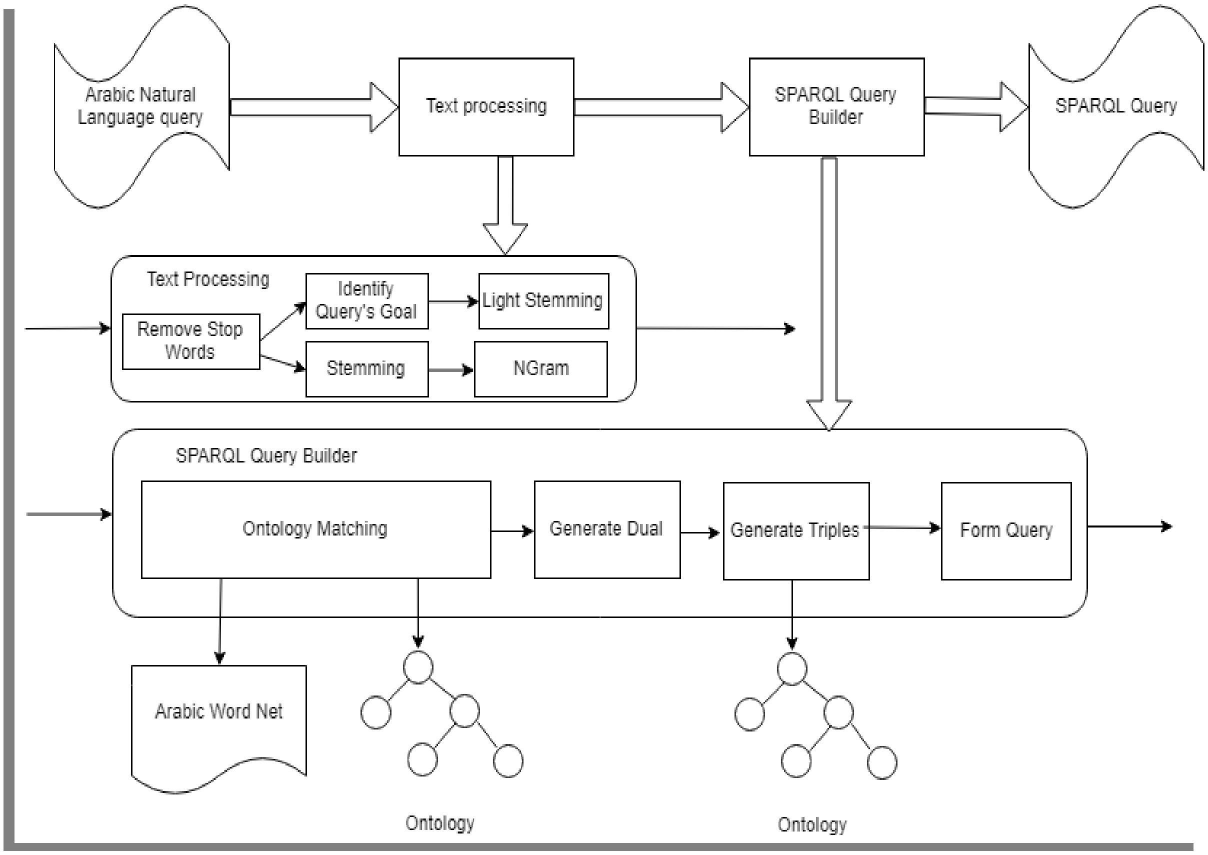
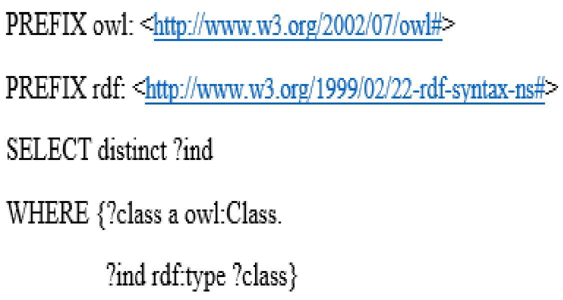
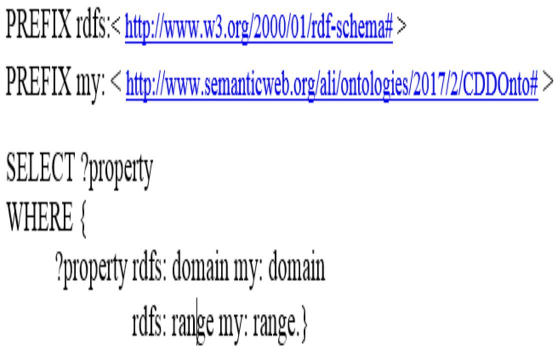


Figure 7: The approach of translating Arabic natural language query to SPARQL query.



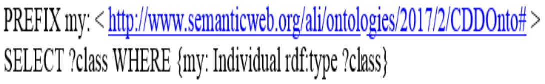
```
PREFIX owl: <http://www.w3.org/2002/07/owl#>

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

SELECT distinct ?ind

WHERE {?class a owl:Class.

          ?ind rdf:type ?class}
```

Figure 8: Query to get individuals in the ontology.



```
PREFIX rdfs:< http://www.w3.org/2000/01/rdf-schema# >
PREFIX my: < http://www.semanticweb.org/ali/ontologies/2017/2/CDDOnto# >

SELECT ?property
WHERE {
     ?property rdfs: domain my: domain
          rdfs: range my: range.}
```

Figure 9: Query to get object property between two classes.



```
PREFIX my: < http://www.semanticweb.org/ali/ontologies/2017/2/CDDOnto# >
SELECT ?class WHERE {my: Individual rdf:type ?class}
```

Figure 10: SPARQL query to get the type of an individual.

6.10. Processing conjunction and disjunction

The previous approach fails when the query contains a conjunction or disjunctions as the following example: ما هو الداء الذي له عرض حمى و بوليبات أنفية i.e. what is the disease that has a symptom of fever and nasal polyps? (1). In order to solve it we do not delete the conjunction or disjunction letters which are 'و' i.e. 'and' and 'أو' i.e. 'or' when deleting the stop words. Then we form the query and concatenate the triples using the conjunctions and disjunctions.

7. EVALUATION OF CDDONTO

In this section, we evaluate the ability of the ontology in representing all terms, properties, and relations through ontology querying using description logic query, which is a standard Protégé plugin and the SPARQL query to verify and validate the ontology in accordance to competency questions. In addition to verify it with the help of a domain expert since we gathered 22 real case for patients then order doctor to diagnose it. After that, compare it with the result of the ontology.

7.1. Ontology evaluation through ontology querying

We present three querying examples, which answer the main questions that are asked in the development process of the ontology.

Example 1:

- The question: what are the diseases that have the symptom إسهال?
- DL_Query: مرض and عرض_له value إسهال.
- Figure 11 shows the result of DL_Query and illustrates the individuals of the disease class.

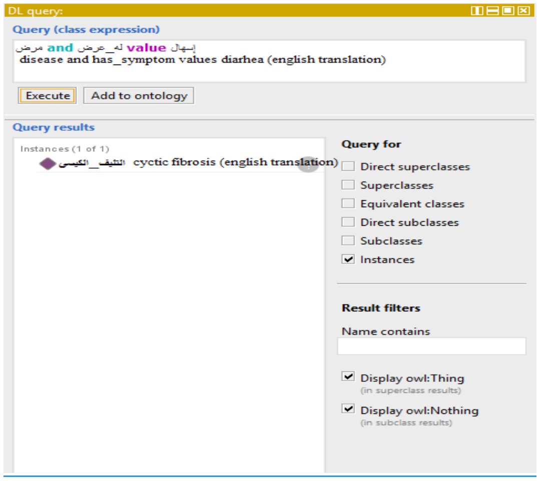
Example 2:

- The question: What are the symptoms of cystic fibrosis disease?
- DL_Query is: عرض and عرض_لـ value التليف_الكيسي.
- Figure 12 shows the result of DL_Query and illustrates the individuals of symptom class.

Example 3:

In this example, we show a SPARQL

- The question: what are the symptoms of cystic fibrosis disease?
- Figure 13 shows SPARQL query and figure 14 shows the result of the SPARQL query, which illustrates the symptom's individuals. That means all these symptoms are symptoms of the cystic fibrosis disease.



مرض and له_عرض value إسهال
disease and has_symptom values diarhea (english translation)

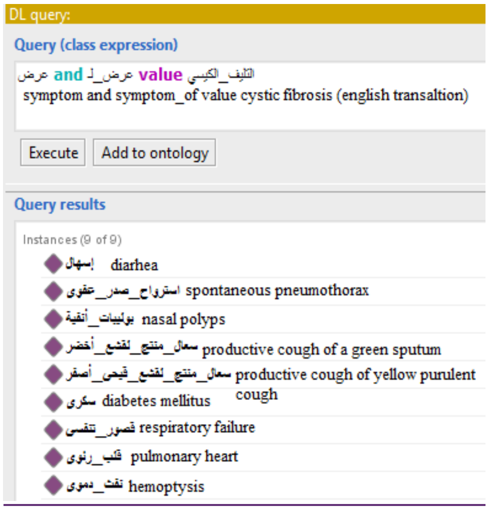
Instances (1 of 1)
التليف_الكيسي cyctic fibrosis (english translation)

Query for
☐ Direct superclasses
☐ Superclasses
☐ Equivalent classes
☐ Direct subclasses
☐ Subclasses
☑ Instances

Result filters
Name contains

☑ Display owl:Thing
(in superclass results)
☑ Display owl:Nothing
(in subclass results)

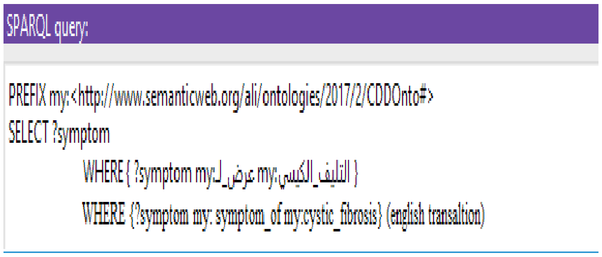
Figure 11: Individuals of a disease.



عرض and عرض_لـ value التليف_الكيسي
symptom and symptom_of value cystic fibrosis (english transaltion)

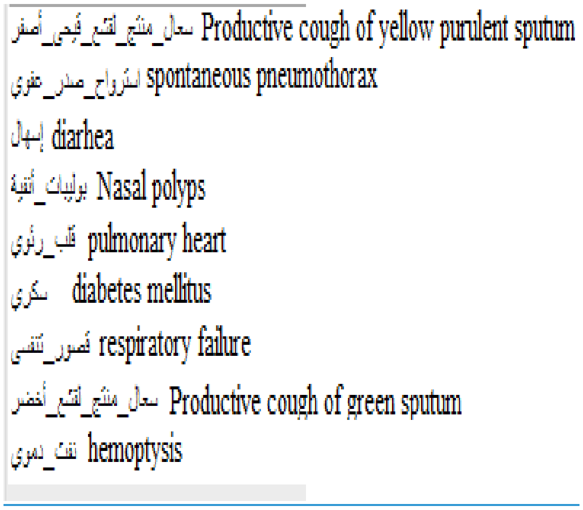
Instances (9 of 9)
إسهال diarhea
استرواح_صدر_عفوى spontaneous pneumothorax
بوليبات_أنفية nasal polyps
سعال_منتج_لقشع_أخضر productive cough of a green sputum
سعال_منتج_لقشع_قيحى_أصفر productive cough of yellow purulent cough
سكرى diabetes mellitus
قصور_تنفسى respiratory failure
قلب_رئوى pulmonary heart
نفث_دموى hemoptysis

Figure 12: Individuals of cystic fibrosis's symptom.



```
PREFIX my:<http://www.semanticweb.org/ali/ontologies/2017/2/CDDOnto#>
SELECT ?symptom
        WHERE { ?symptom my:عرض_لـ my:التليف_الكيسي }
        WHERE {?symptom my: symptom_of my:cystic_fibrosis} (english transaltion)
```

Figure 13: SPARQL query.



سعال_منتج_لقشع_قيحي_أصفر Productive cough of yellow purulent sputum
استرواح_صدر_عفوي spontaneous pneumothorax
إسهال diarhea
بوليبات_أنفية Nasal polyps
قلب_رئوي pulmonary heart
سكري diabetes mellitus
قصور_تنفسي respiratory failure
سعال_منتج_لقشع_أخضر Productive cough of green sputum
نفث_دموي hemoptysis

Figure 14: SPARQL query result.

7.2. Quality Evaluation through Disease Examples

To evaluate the quality of the CDDOnto we chose a disease example to check if the ontology represents terms, properties and relations of a disease sample. Figure 15 shows example of acute bronchitis (التهاب القصبات الحاد) disease, which is an individual of disease (مرض) class, and has nine symptoms and two investigations.

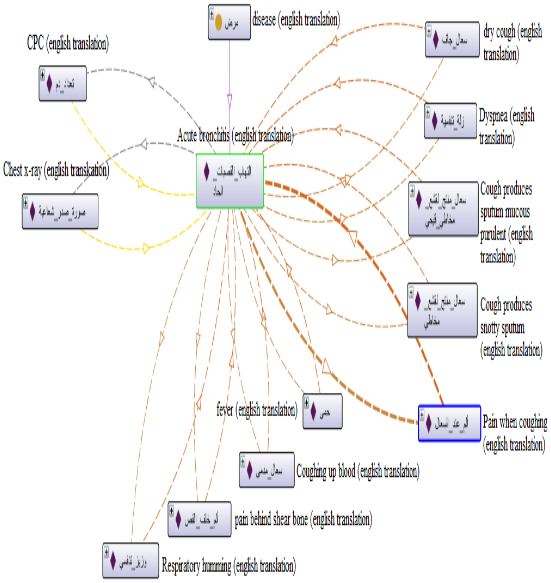

مرض
disease (english translation)
CPC (english translation)
تعداد_دم
dry cough (english translation)
سعال_جاف
Dyspnea (english translation)
زلة_تنفسية
Acute bronchitis (english translation)
التهاب_القصبات_الحاد
Chest x-ray (english transkation)
صورة_صدر_شعاعية
Cough produces sputum mucous purulent (english translation)
سعال_منتج_لقشع_مخاطي_قيحي
Cough produces snotty sputum (english translation)
سعال_منتج_لقشع_مخاطي
fever (english translation)
حمى
Pain when coughing (english translation)
ألم_عند_السعال
Coughing up blood (english translation)
سعال_مدمي
pain behind shear bone (english translation)
ألم_خلف_القص
Respiratory humming (english translation)
وزيز_تنفسي

Figure 15: Object properties of acute bronchitis disease.

7.3. Ontology evaluation with real cases

In this step, we evaluate the ontology using 22 real cases for patients. Then order a specialist doctors to diagnose them. After that, we compare it with the results we gathered from the ontology. The ontology shows an accuracy of 72.7%.

8. CONCLUSION

In this paper, we proposed an Arabic ontology-based approach for diagnosing chest diseases. We focused on the process of building an ontology. The ontology content is related to the medical domain and It was gathered from [7] and from a domain expert. The ontology provides a knowledge base of chest diseases, symptoms, and investigations. The proposed ontology is evaluated with a help of domain experts and showed an accuracy of 72.7%.

Then we developed a natural language processing model, which enables users query the ontology using their Arabic natural language and get the desired knowledge.

In the future, we intend to save the translation results of the queries, then using data mining techniques to get the translation based on the closest case to the case at hand.

9. ACKNOWLEDGEMENTS

This work was supported by the student grant CTU SGS16/253/OHK3/3T/14.

REFERENCES

[1] ALFONSE, M., AREF, M., & SALEM, A.-B. (2013). An Ontology Based Cancer Diseases Diagnostic Methodology. European computing conference (pp. 95-99). Greece: WSEAS Press.

[2] Wang, h., & Tansel, A. (2013). Composite Ontology-Based Medical Dianosis Desision Support System Framework. Communication.

[3] Tris, C. R. (2009). Ontology Based Diagnosis and Personalization of Medical. Universitat Rovira i Virgili.

[4] Jayaratne, L. (2015). Ontology Based Approach for Diagnosis in Personalized Medicine. *GSTF Journal on Computing Vol. 4 No .2.*

[5] Joshi, A., & Dhariwal, D. (2013). Text And Ontology Driven Clinical Decision Support System. *AMIA.*

[6] Baraka, R., & El-Askary, M. (2015, August). An Ontology-Based Approach for diagnosis Date Palm Diseases. Gaza, Palestine: Islamic University of Gaza.

[7] AlAgha, I. (2015). Using Linguistic Analysis to Translate Arabic Natural Language Queries to SPARQL. Gaza: Cornell University Library.

[8] Shaik, S., Kanakam, P., Hussain, M., & Suryanarayana. (2016). Transforming Natural Language Query to SPARQL for Semantic Information Retrieval. International Journal of Engineering Trends and Technology.

[9] Sander, M., Waltinger, U., Roshchin, M., & Runkler, T. (2014 Septemper, Septemper). Ontology-Based Translation of Natural Language Queries to SPARQL. Natural Language Access to Big Data.

[10] Ngomo, A.-C. N., Bühmann, L., Unger, C., Lehmann, J., & Gerber, D. (2013 May). Sorry, I don't speak SPARQL – Translating SPARQL Queries into Natural Language. Proceedings of the 22nd international conference on World Wide Web (pp. 977-988). ACM.

[11] Pradel, C., Ollivier, & Hernandez, N. (2013). Natural language query interpretation into SPARQL using patterns. Fourth International Workshop on Consuming Linked Data, (pp. 1-12).

[12] kim, J.-D., & Cohen, K. (2013). Natural Language Query Processing for SPARQL generation a Prototype System for SNOMEDCT. Proceding of biolink, (pp. 32-38).

[13] Haslett, C., Simmons, D. E., & R. chilvers, E. (2002). Dividson Principles and Practise of Medicine. Churchill Livingstone.

[14] Khondoker, R. M., & Muller, P. (2010). Comparing Ontology Development Tools Based On an Online Survey. Proceedings of the World Congress on Engineering. London: IAENG Online Journal.

[15] Noy, N., & McGuinness, D. (2001). Ontology Development 1010: A Guide to Create Your First Ontology. California: Stanford University.

[16] Sookocheff, K. (2015, 7 25). Modeling Natural Language with N-Gram Models. Retrieved from Kevin Sookocheff: https://sookocheff.com/post/nlp/n-gram-modeling/

[17] Taghva, K., EL-Khoury, J., & Coombs, j. (2005). Arabic Stemming without a root dictionary. Las Vegas, Nevada, United States.