Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 83 (2016) 607 – 614
Procedia Computer Science

The 7th International Conference on Ambient Systems, Networks and Technologies (ANT 2016)

Cross-Domain Semantic Web Model for Understanding Multilingual Natural Language Queries: English/Arabic Health/Food Domain Use Case

Ahmed Al-Nazer, Saeed Albukhitan, Tarek Helmy*

Information and Computer Science Department, College of Computer Science & Engineering, King Fahd University of Petroleum & Minerals, Dhahran 31216, Mail Box 413, Saudi Arabia, [alnazera, albokhitan, helmy]@kfupm.edu.sa

Abstract

With the growth of the Semantic Web and its applications, the need to use it in different languages, such as Arabic, is becoming more important. Two of the challenges with the Semantic Web technologies are the lack of multilingual support and the complexity of integrating multiple ontologies used by this technology. The objective of this paper is to present efforts that will help users who use the Arabic language to ask natural language questions and then get their semantic representation in SPARQL that allows them to be executed and get the relevant semantic results. This natural language interface makes more use of the cross-domain ontologies and hence improves the understanding of their inquiries, which is needed in some critical domains such as health and food where precise advice is essential. The approach we followed is multilingual and overcomes the limitations in the published relevant systems. With the proposed approach, users who speak Arabic can use the widely published ontologies in English without concern for the translation of their questions. The proposed approach will take care of matching the entered questions with the relevant ontologies to produce their semantic Web queries. The proposed approach has been implemented and empirically evaluated. The experimental results are promising, which will help in improving the awareness and usage of the Semantic Web by different lingual and cultural users.

Peer-review under responsibility of the Conference Program Chairs

Keywords: Semantic Web; SPARQL; Arabic Natural Language; Question Answering; Semantic Web Service.

* Corresponding author. Tel.: +966-3-860-1967; fax: +966-3-860-2174.
E-mail address: helmy@kfupm.edu.sa
On leave from College of Engineering, Tanta University, Egypt

1877-0509
Peer-review under responsibility of the Conference Program Chairs
doi:10.1016/j.procs.2016.04.138

1. Introduction

The Semantic Web is an extension of the normal Web where data are linked and reasoned. Semantic Web technologies help in building knowledge bases that contain linked vocabularies along with rules that organize the relationship between these concepts. While there are many applications and domains that have used Semantic Web technologies, there are growing challenges when it comes to using these technologies, especially when it requires average users to link different languages and domains. A lot of Semantic Web resources and data sets are available in common languages, such as English, while users from different languages are challenged to utilize this wealth of information on the Web. Although there are some efforts to translate the queries on the common search engines, translation wouldn't work on the Semantic Web because of the nature of the technology that relies on the meaning of the question. Thus, we address this challenge in this paper by proposing a model that deals with this barrier in a better way to allow users from different languages, such as Arabic, to use the available resources in English. Correlating different domains is one of the challenges once we start looking into the Semantic Web repositories, as each domain has its own experts who are focusing on the domain vertically. Because there is a need to deal with the domains horizontally and manage cross-domain questions, we address this challenge through this paper, which deals with critical domains. In this paper, health and nutrition domains were chosen as a case study because they are critically important. Also, there is a growing demand on the Web for answers to health-related questions, while the common search engines still depend on the keywords and hence mislead the end-users. In the health and nutrition domains, we focus on the relations between "*diseases*," "*body parts*" and "*body functions*" with the food and nutrients.

The Natural Language Interface (NLI) deals with the Natural Language (NL) queries and questions. It takes care of the complexities behind the used technology and eases the input required by the common users. We implement an NLI that can help in reaching out to the Semantic Web repositories and ontologies to produce the matching Semantic Web queries to the user's questions. This is not limited to the specific language the ontology is in; rather, it extends to be used in cross-language use cases. In this paper, we address these challenges with the objective to have an easy-to-use tool that can convert the natural language queries into Semantic Web queries using the relevant technology, namely, Simple Protocol and RDF Query Language (SPARQL). We aim to translate and match the NL input into the relevant formal query written in SPARQL. This query can then be executed in the data store to get the matching Semantic Web results. Any SPARQL query consists of a number of triples, in the form of subject-predicate-object, where the query reasoning engine matches the triples of the SPARQL query with the stored Resource Description Framework (RDF) triples in the knowledge base. These knowledge bases are created in an earlier phase during the annotation process.

The rest of this paper is organized as follows. Section 2 reviews the existing related work on the NLI for the Semantic Web and their Arabic language support. Section 3 presents the ontologies we have used in our use case. Section 4 presents the architecture of the proposed approach. Section 5 discusses the process of semantic query manipulation. Section 6 presents the experimental results and discussion. Finally, we summarize the paper and highlight the future work directions in Section 7.

2. Related Work

In this section, we present a literature review on the related work to the NLI for the Semantic Web and their Arabic language support. Let's discuss first the support for Arabic language in the Semantic Web technologies. Based on our findings, there is limited work on the use of the Semantic Web technologies to build applications in the Arabic language. Yet, there is some work on the development of Arabic ontologies, which can be found in the literature[1,2,3]. There are also some works related to one stage in the Semantic Web, namely the entity extraction, in Arabic using ontologies[4,5]. There are also a number of papers that employ the ontology representation for Islamic knowledge[6,7,8]. In addition, there are some publications that use ontology on cross-language information retrieval[9, 10]. Most of the use for ontologies in the information retrieval is focusing on the query expansion where the results were pulled from the Web[11,12]. Our work takes a different route, where we address the query interface based on the existing ontologies for RDF stores that are used in the Semantic Web to hold the knowledge bases. There is a noticeable growth in the efforts to build Arabic ontologies. A good example is the Arabic Ontology project[13], which has a goal to build an Arabic version of WordNet. Also, there is a work on building Quranic Ontology that uses Quranic concepts to represent the knowledge and relationships between Quranic concepts using the formal RDF triple representation[14]. Some other research work emphasizes creating DBpedia that uses the Linked Data concept in Arabic content enrichment[15,16]. In this paper, we focus on NLI that can help the Arabic community and utilizes the existing Arabic content by translating the NL query into a Semantic Web query, which can be executed on top of these Arabic ontologies.

On the English NLI, there is a good number of works that have been published about query interfaces for Semantic Web data sources and ontologies. An example is AquaLog system[17] that transforms the questions and is built on top of Linked Data. It uses reasoning technologies in the Semantic Web to learn and build new patterns. One limitation of AquaLog is that it derives the queries from single ontologies only. Another example is NLPReduce NLI[18] that creates a Semantic Web query out of the user's query by correlating the terms and trying to find a triple relationship. This system has a disadvantage, which is its dependence on predefined grammar used to build the triples and identify the concepts, as well as the relationships. This makes it more difficult to deal with more complicated queries that do not match with the predefined templates. To overcome this limitation, some work suggests using linguistic parsing, instead of the predefined roles, to recognize the terms in the query and their relationships. This will allow the Natural Language Interface to deal with more complex queries. An example is found in PANTO[19], which presents an NLI that uses linguistic parsing to understand the user's query, identify the related terms with their relationships, and then construct the Semantic Web triples. Another example of the linguistic parsing use is found in Unger et al.[20], which constructs a SPARQL pattern that relates the user's question and maps it to the domain concepts. These approaches that worked in English, with some limitations, do not necessarily work in the Arabic language, as Arabic is more complex and has more complicated morphology. In this paper, we are going to address these challenges. Finally, there is one recent publication about AR2SPARQL[21] that tries to address these challenges by proposing a QA system that helps in translating the Arabic queries into Semantic Web triples using linguistic reasoning. ARSPARQL lacks the manipulation of multiple ontologies and is very limited to a single ontology.

3. Architecture of the Proposed Approach

The architecture of the proposed approach is to address the transformation of the natural language query to the SPARQL semantic query based on multilingual and cross-domain ontologies. The transformation process consists of a number of steps starting with linguistic processing, then semantic processing, and finally SPARQL query generation via interacting with the ontologies. This architecture is detailed in Figure 1. First, it accepts the NL query through its interface, processes it, and then finally produces the SPARQL query that can be executed on the repository of knowledge bases. Based on the ontology selection, the Name Entity List gets generated and all name entities are extracted to help in processing the query. The Name Entity List consists of all concepts in the knowledge base that is built based on the selected ontologies in addition to the predefined relationships between these concepts.

The process of analyzing the NL query and generating the SPARQL query consists of three main phases: (1) query preprocessing, (2) query mapping, and (3) SPARQL query generation. Let's start from the time the query is entered into the system. It is linguistically processed and analyzed to understand it, then mapped to its semantic representation. The system detects the query language first, as our system is cross-lingual, and in this time it supports both English and Arabic. Each language has its own way of processing, as it has special characteristics that make the understanding of the natural language query different. Then, we do an extra step to spell-check the query and make any necessary corrections, as this will impact the next steps. After that, the terms of the query are tokenized and isolated to take them one by one into the next steps. It is not necessary for tokenizing to be word by word, as we will see later; if we have ambiguity in understanding the query, then we try to correlate the words next to each other. After that, we classify the tokens into their Part-Of-Speech (POS) to identify the verbs from the nouns and other formats like adjectives and adverbs. We use the Stanford Log-linear POS Tagger[†] that is based on the Penn Treebank

[†] http://nlp.stanford.edu/software/tagger.shtml

POS[‡] Tags. For the Arabic language part of speech tagging, we use extra tags by Stanford known as DTNN that handles special Arabic characters like the words that start with "ال" which is pronounced as "al." Next, we remove the unwanted words that add noise to the processing, such as "is," "are," etc. Finally, the query undertakes the normalization and stemming steps to ensure the correct mapping in the next step.

After preprocessing the query, we start mapping it into the semantic form that can help in generating the SPARQL query. There is a close correlation between mapping the query and generating the SPARQL form, as the generation is made straightforward based on the mapping results. First, the question is classified to make sure we construct the correct SPARQL query. A question that is asking for quantities of daily intake of Vitamin C is different than a question asking about listing the benefits of eating an apple. It is essential to distinguish each question in order to construct the appropriate SPARQL query. Then, we identify the measurements that are listed in the question, if any. After that, we start the semantic mapping between the terms in the question, and the name entities exist in the ontology knowledge base that was constructed previously. This will help to identify the concepts that correlate to the terms in the query. Finally, the relationships between the terms are identified by matching them to the relationships that exist in the ontology knowledge base. This step is critical and required to understand the whole query and correlating the words with each other.

The final step is the SPARQL query generation, which is the assembly of all that has been identified in the previous steps. The generator has the flexibility to accept all combinations and then generates the appropriate SPARQL query based on the reference ontology. After the SPARQL query is generated, it is exported to the user who can execute it over the knowledge base and get the matching RDF results. The basic form of any SPARQL query has two parts: SELECT clause and WHERE clause. The SELECT clause sets the parameters that will result after executing the query, while the WHERE clause sets the filters that are used to select the results. These clauses contain a triple that represents the semantic information, <*subject*, *predicate*, *object*>. These parameters can be written as Uniform Resource Identifier (URI) representing a variable or can be a value. In the SPARQL query generator, we set the values where we have them from the input query, and we set the variables where there is unknown information or questioned information. In the next section, we illustrate a full example that highlights this.

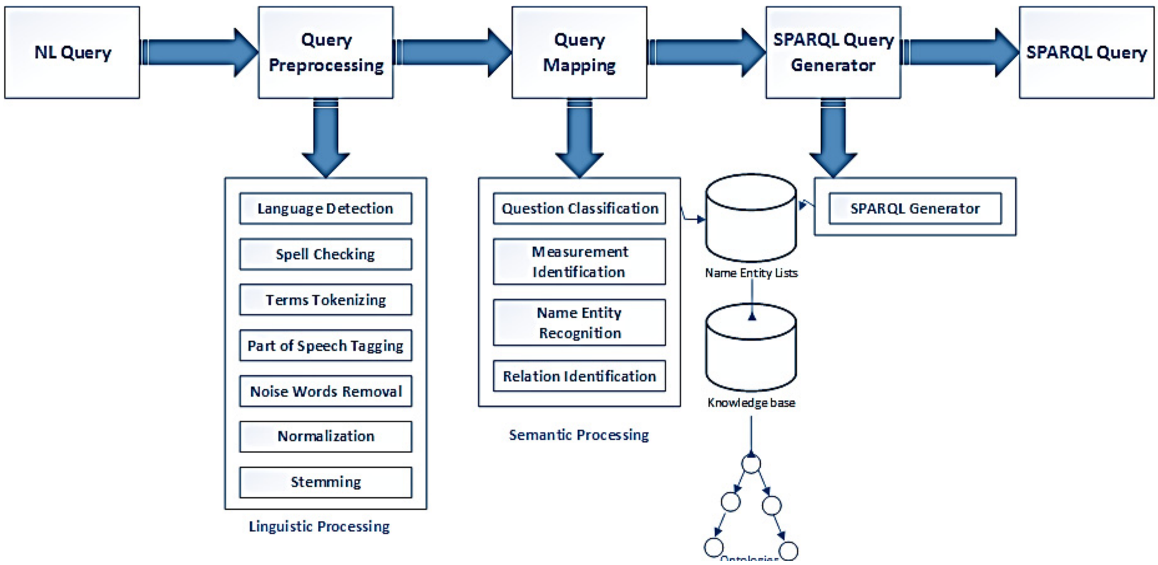

Figure 1: The Architecture of the Proposed Approach

‡ https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html

4. Used Ontologies

In this section, we introduce the ontologies we have used in our use case. Ontology is a network representation of knowledge for each domain that consists of concepts that have types, properties and relationships between each other. In our experiment, we have integrated different ontologies related to health and food. We went through three main steps in our work with the ontologies: (1) ontology selection, (2) ontology integration, and (3) ontology multilingualism.

For ontology selection, we used the Semantic Diet[§] food ontology, which is aligned with the USDA food database for both food and nutrition ontologies. We adapted the human disease ontology (DO)[**] that contains information about the diseases to fit and integrate with our ontologies. Diseases are linked to our body through our body's parts and our body's functions. For this, we created our own ontologies that include the human body and its functions.

For ontology integration, we have constructed an integration ontology, which integrates the health ontologies (*disease*, *bodyParts*, *bodyFunctions*) with Food ontologies (*Fooditem* and *Nutrient*). We have done this through creating relationships between the domains that help in reasoning later on. The integration's relationships are the following: *causeTo*, *preventFrom*, *appearIn*, *treatFrom*, *hasNegativeEffectOn* and *hasPositiveEffectOn*.

For ontology multilingualism, the multilingual property was used to produce a multilingual ontology that covers the English and Arabic languages. With the objective to have integrated multilingual ontologies, we take an input for this process as the English ontologies and we produce the output as the integrated multilingual ontologies using either (i) ontology-to-ontology mapping or (ii) agnostic ontology, which plays the role of a bridge between existing ontologies. To achieve this output, we use one of the three methodologies: (a) automatically align the ontologies by using a translation service or mediator such as Wikipedia, (b) manually align the ontologies, or (c) semi-automatically align the ontologies, i.e., with human guidance. We followed this process in the ontologies we have used, as explained in the next section. Figure 2 shows illustrations of the used ontologies.

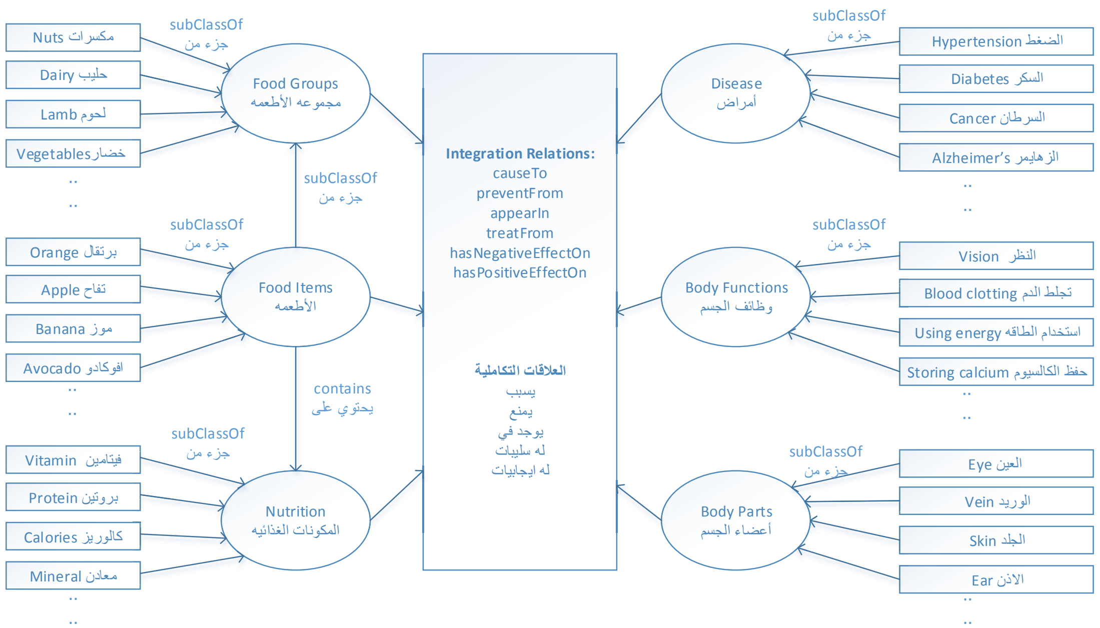

Figure 2: Used Food and Health Ontologies

[§] http://semanticdiet.com/
[**] http://disease-ontology.org/

5. Query Processing Example

The semantic query manipulation process has a number of steps that have been discussed in Section 4. In this section, we show an example of processing an NL query. In this example, the query entered by the user is "ما هي الأطعمة التي تفيد مرضى الضغط؟" which corresponds to the English query "what food that benefits the people with hypertension?". In this query, the preprocessing steps take place and produce the following output: "الأطعمة" /DTNN, "تفيد" /VBP, "الضغط"/DTNN. Also, the following terms will be used to classify the question in the next step: ما/WP هي/PRP, which is "List Question." After finding that there are no measurement terms in the question, the semantic processing tries to correlate the terms to the concepts from the Name Entity List. Here, we find the following: الأطعمة (concept: FOOD), الضغط (instance: DISEASE: HYPERTENSION). After that, we try to find the relation between these concepts, and here we find: تفيد (relation: hasPositiveEffectOn). This will be translated to the SPARQL query generator to produce a corresponding SPARQL query that lists the food that has a positive effect on the hypertension disease. The following SPARQL query in Figure 3 will result after the generation is completed. This SPARQL query can then be executed over the knowledge base and retrieve semantic results. Figure 4 shows example results that are returned after executing the SPARQL query. These results are then evaluated to determine whether or not they are correct and match the user's expectation, as explained in the next section.

```
select  ?document_text ?document_url ?health ?foodItem ?hasText  where {
?relation rdf:type integration:Relation .?relation integration:appearsIn
?sentence  .?document integration:containsSentence  ?sentence .?sentence
integration:content ?document_text . ?document integration:hasURL ?document_url
.?relation integration:preventTo ?health .?relation integration:preventFrom
?foodItem .?relation integration:hasText ?hasTxt . bind(str(?hasTxt) as
?hasText).?relation integration:hasTextFrom ?hasTxtFrom . bind(str(?hasTxtFrom)
as ?hasTextFrom).?relation integration:hasTextTo ?hasTxtTo .
bind(str(?hasTxtTo) as ?hasTextTo).?health rdfs:label ?health_lbl . ?foodItem
rdfs:label ?foodItem_lbl . filter(lang(?health_lbl) = "en") .
filter(lang(?foodItem_lbl) = "en") . bind(str(?health_lbl) as ?health_label) .
bind(str(?foodItem_lbl) as ?foodItem_label)
.filter(strstarts(str(?health),"http://www.kfupm.edu.sa/ontology/health/"))
.filter(strstarts(str(?foodItem),"http://www.kfupm.edu.sa/ontology
/food/foodItem/")) .filter(strstarts(str(?health),"http://www.kfupm.edu.sa
/ontology/health/disease/I_DOID_11714")). }
```

Figure 3: The output of the SPARQL Generator

Document_text	Document_url	Health	FoodItem	HasText
"Research has found that ginger may be beneficial in the fight against diabetes."@en	"http://www.healthdiaries.com/eatthis/ginger"^^xsd:string	<http://www.kfupm.edu.sa/ontology/health/disease/I_DOID_11714>	<http://www.kfupm.edu.sa/ontology/food/foodItem/I11216>	"fight against"
"Research has found that ginger may be beneficial in the fight against diabetes."@en	"http://www.healthdiaries.com/eatthis/blood-sugar"^^xsd:string	<http://www.kfupm.edu.sa/ontology/health/disease/I_DOID_11714>	<http://www.kfupm.edu.sa/ontology/food/foodItem/I11216>	"fight against"
"Research has found that ginger may be beneficial in the fight against diabetes."@en	"http://www.healthdiaries.com/eatthis/archives.html"^^xsd:string	<http://www.kfupm.edu.sa/ontology/health/disease/I_DOID_11714>	<http://www.kfupm.edu.sa/ontology/food/foodItem/I11216>	"fight against"
"Studies have found that evening primrose oil may also help protect from diabetic neuropathy, a nerve disorder often affecting people with diabetes that causes tingling, pain, numbness, and other symptoms in the legs and feet."@en	"http://www.healthdiaries.com/eatthis/8-health-benefits-of-evening-primrose-oil.html"^^xsd:string	<http://www.kfupm.edu.sa/ontology/health/disease/I_DOID_11714>	<http://www.kfupm.edu.sa/ontology/food/foodItem/I04034>	"may also help protect"
"I have read above blog of 8 health benifit of barley grass powder.It contains as 70 different different minerals, numerous vitamins, enzymes and antioxidants and protein.it prevent cancer,diabetes and balance colestrol and more benifit provide in our body.so i suggested every one take this one in our increase long lifestyle."@en	"http://www.healthdiaries.com/eatthis/8-health-benefits-of-barley-grass.html"^^xsd:string	<http://www.kfupm.edu.sa/ontology/health/disease/I_DOID_11714>	<http://www.kfupm.edu.sa/ontology/food/foodItem/I20004>	"prevent"
"A review of 18 studies, involving 450,000 people, published in Archives of Internal Medicine , found that each additional cup of coffee consumed per day lowered the risk of diabetes by 7%."@en	"http://www.healthdiaries.com/eatthis/6-health-benefits-of-decaf-coffee.html"^^xsd:string	<http://www.kfupm.edu.sa/ontology/health/disease/I_DOID_11714>	<http://www.kfupm.edu.sa/ontology/food/foodItem/I14201>	"lowered the risk of"

Figure 4: The output of executing the SPARQL Query

6. Experimental Results and Evaluation

In order to validate the functionalities and the accuracy of the proposed approach, we have run different experiments. In this section, we summarize one of the experiments that we have performed on the proposed approach. We came across a number of challenges that we talk about in the following section. We have collected 389 NL questions from different sources on the Web, such as specialized, domain-specific websites, and from surveys we have conducted on the local users. Then, we have classified these questions based on the main query in the question (whether it is in the food or health domain), and more specific questions based on our ontologies. Also, we categorized these questions based on their type. Table 1 below shows the summary of the input.

Table 1: Summary of the Experiment Input

Question Category	Question Type			Total per Category
	List	Quantity	Confirmation	
Food-based questions	59	19	37	115
Nutrition-based questions	31	22	34	87
Disease-based questions	37	19	29	85
Body part-based questions	15	9	23	47
Body function-based questions	19	8	28	55
Total	161	77	151	389

To measure the accuracy of the NL questions understanding, we run the questions through a team to do a manual step in identifying the concepts and the relations in the 389 questions. That will be a basis for the accuracy measurement of the proposed approach. We used the following matrices to measure the performance of our solution: Precision, Recall and F-measure. The Precision (P) assesses the accuracy and is calculated by dividing the correctly identified concepts by the total number of concepts found (both right and wrong). The Recall (R) assesses the coverage and is calculated by dividing the correctly identified concepts by those that were found manually. Finally, F-measure is the trade-off and is calculated by multiplying 2 times the product of Precision and Recall divided by the sum of the Precision and Recall. Mathematically, the f-measure formula is: *F-measure = 2 * (P*R) / (P+R)*.

Table 2 below shows the results of measuring these matrices to measure the accuracy of the query understanding. As shown, the resulting Precision is high, on average (0.89 out of 1.0), which projects high accuracy, while the Recall needs improvement, as it is 0.79, which reflects the need to add more concepts into the knowledge base. We also noticed that the disease-based questions have the lowest Precision, 0.82, which indicates the varieties of the disease names that the questions contain. The results are not compared to other experiments because of the fact that other systems are either monolingual or supporting one ontology only. We plan to have a common data set in the future that can help in comparing our approach to others in specific features such as serving single ontology.

Table 2: Summary of the Results

Question Category	Measure		
	Precision	Recall	F-Measure
Food-based questions	0.92	0.86	0.89
Nutrition-based questions	0.90	0.78	0.84
Disease-based questions	0.82	0.77	0.79
Body part-based questions	0.91	0.81	0.86
Body function-based questions	0.91	0.72	0.80
Average	0.89	0.79	0.84

While mapping the Arabic terms in the NL query into the corresponding semantic concepts, there were a number of challenges. First, the Arabic language has different formats of some of its letters; for example, the letter "أ" has many formats: "أ", "إ", "ا", etc. We have to deal with all formats of the Arabic letters in our processing, and that is why we used stemming as a powerful tool that returns the word to its root. The second challenge was to deal with the phrases that consist of more than one word. This is very commonly used in the Arabic language. A good example is "يساعد على الوقاية" which means "helps in protecting." Words can have contradictory meanings based on the combination. The third challenge is the ambiguity that we face when we try to match a term in the natural language query to the concept in the knowledge base, as some words have multiple meanings; for example, "السكر" means the diabetes disease and the sugar food or nutrition. We have to combine the question's terms to determine the closest matching of the terms. In certain cases, we have to construct different queries based on different meanings and assign an accuracy percentage for each query. In additional to our reference to WordNet, we populate the synonyms of some terms, especially the relations between concepts, to minimize the ambiguity and increase the coverage. One last effort we have addressed is to have better matching between the NL query and the concepts in the knowledge base. Before populating the Name Entity List from the knowledge base, we processed the concepts via normalization and stemming to ensure that we could match them with the terms found in the query.

7. Conclusion and Future Work

In this paper, we propose a new methodology to handle cross-lingual multi-ontologies that helps in transforming the NL query into the SPARQL query that can be executed over the RDF knowledge base to get the semantic results. This will close the gaps that we have identified through our survey of the lack of such approaches that work for Arabic and English on more ontologies. This paper shares the projection of this methodology on the health and food domains with some experimental results that look promising. As a future work, we plan to publicize the developed work to first test our approach and then to have more use cases other than food and health. In addition, we will compare different algorithms in the linguistic analysis to identify the best-fit algorithm for our use case. This requires more data to be collected and built that can help in assessing the accuracy of the proposed approach.

Acknowledgements

Thanks to King Fahd University of Petroleum & Minerals for supporting this work through project no. IN141038.

References

1. Hazman, M., S.R. El-Beltagy, and A. Rafea. Ontology learning from domain specific web documents. International Journal of Metadata, Semantics and Ontologies. 4(1): p. 24-33, 2009.
2. Ishkewy, H., H. Harb, and H. Farahat. Azhary: An Arabic Lexical Ontology. International Journal of Web & Semantic Technology. 5(4), 2014.
3. Jarrar, M. Arabic ontology engineering-challenges and opportunities. in Proceedings of the 2011 International Conference on Intelligent Semantic WebServices and Applications: ACM, 2011.
4. Abouenour, L., K. Bouzoubaa, and P. Rosso. Using the Yago ontology as a resource for the enrichment of Named Entities in Arabic WordNet. in Editors & Workshop Chairs, 2010.
5. Zaidi, S., M. Laskri, and A. Abdelali. Arabic collocations extraction using Gate. in Machine and Web Intelligence (ICMWI), 2010 International Conference on: IEEE, 2010.
6. Abdelnasser, H., et al. Al-Bayan: An Arabic Question Answering System for the Holy Quran. ANLP 2014. p. 57, 2014.
7. Iqbal, R., A. Mustapha, and Z.M. Yusoff. An experience of developing Quran ontology with contextual information support. Multicultural Education & Technology Journal. 7(4): p. 333-343, 2013.
8. Khan, H.U., et al. Ontology Based Semantic Search in Holy Quran. International Journal of Future Computer and Communication. 2(6): p. 570-575, 2013.
9. Hattab, M., et al. Addaall Arabic Search Engine: Improving Search based on Combination of Morphological Analysis and Generation Considering Semantic Patterns. in The second International Conference on Arabic Language Resources and Tools, Cairo, Egypt, 2009.
10. Tazit, N., et al. Semantic internet search engine with focus on Arabic language. The 1st International Sysmposium on Computers and Arabic Language & Exhibition, 2007.
11. Mahgoub, A.Y., et al. Semantic Query Expansion for Arabic Information Retrieval. ANLP 2014. p. 87, 2014.
12. Fader, A., L.S. Zettlemoyer, and O. Etzioni. Paraphrase-Driven Learning for Open Question Answering. in ACL (1), 2013.
13. Jarrar, M. Arabic ontology engineering-challenges and opportunities. in Proceedings of the 2011 International Conference on Intelligent Semantic WebServices and Applications: ACM, 2011.
14. Dukes, K., E. Atwell, and N. Habash. Supervised collaboration for syntactic annotation of Quranic Arabic. Language resources and evaluation. 47(1): p. 33- 62, 2013.
15. Al-Feel, H. A Step towards the Arabic DBpedia. International Journal of Computer Applications. 80(3): p. 27-33, 2013.
16. Bahanshal, A.O. and H.S. Al-Khalifa. Toward Recipes for Arabic DBpedia. in Proceedings of International Conference on Information Integration and Web-based Applications & Services: ACM, 2013.
17. Lopez, V., M. Pasin, and E. Motta. Aqualog: An ontology-portable question answering system for the semantic web, in The Semantic Web: Research and Applications Springer. p. 546-562, 2005.
18. Kaufmann, E., A. Bernstein, and L. Fischer. NLPReduce: A "naive" but Domain-independent Natural Language Interface for Querying Ontologies: ESWC Zurich, 2007.
19. Wang, C., et al. Panto: A portable natural language interface to ontologies, in The Semantic Web: Research and Applications Springer. p. 473-487, 2007.
20. Unger, C., et al. Template-based question answering over RDF data. in Proceedings of the 21st international conference on World Wide Web: ACM, 2012.
21. AlAgha, I. and Abu-Taha, A. AR2SPARQL: An Arabic Natural Language Interface for the Semantic Web. International Journal of Computer Applications 125(6):p. 19-27, September 2015.