A COMPUTATIONAL METHOD TO ANSWER SEMANTIC QUESTIONS WITH CONSTRAINTS

Nurfadhlina Mohd Sharef[1], Shahrul Azman Mohd Noah[2], Rabiah Abdul Kadir[1], and Azreen Azman[2]

[1]*Intelligent Systems Research Group, Faculty of Computer Science and Information Technology, University of Putra Malaysia, Malaysia, fadhlina@fsktm.upm.edu.my , rabiah@fsktm.upm.edu.my, azreen@fsktm.upm.edu.my*

[2]*Knowledge Technology Group, Centre of Artificial Intelligence, Faculty of Technology and Information Science, National University of Malaysia, samn@ftsm.ukm.my*

ABSTRACT. Semantic question answering is a discipline that allows the development of smart systems through its reasoning and natural language understanding capability. Knowledge is stored based on representation format such as ontology. However, querying these knowledge requires understanding of the knowledge structure and demands the user to be equipped with the formal query construction skill. This will definitely hinder the development of this area. Therefore this paper addresses the translation model of constraints-typed questions. We focus on the components that relate to constraints-typed question answering and propose a computational model for the translation. These figures indicate promising development in this problem and should encourage more alternative methods in the same direction.

Keywords: Semantic, Question Answering, SPARQL, Natural Language

INTRODUCTION

Semantic question answering (SQA) over the steadily growing amount of semantic data opens possibilities not conceivable before; deep and accurate answering, compared to keyword based matching adopted by information retrieval approach. Through the semantic represented knowledge, reasoning is allowable by connecting and making sense of the content of the knowledge.

However, due to the structured format, the knowledge cannot be benefited by the novice user without mastering the query language thus natural language (NL) question is the best media. On the other hand, the question may require composed information from several sources. Furthermore, variation of question complexity demands different execution strategies and thus traditional federated query approach is unsuitable. Although sources mapping can be performed, this is typically with low automation thus demands high human labor.

Traditional information retrieval approach is insufficient to solve this problem because it cannot exploit the internal structure of the data. Besides, query federation strategy will also fail because it typically distribute the questions into the resources and integrate the answer. This will return very low accuracy because (i) automatic identification of the source to utilize is challenging, (ii) the result may not be suitable for straightforward integration. Therefore, formal query (i.e, SPARQL) needs to be constructed to retrieve the data from the semantic sources.

However, constructing a SPARQL query demands proficiency in the language syntax and mastering the schemes. (Kaufmann & Bernstein, 2007) discovered that casual user preferred an interface that use natural language (NL) compared to those keyword based, partial sentences and graphical. However, processing natural language questions are challenging due to the inherited ambiguity in the language. In the semantic search situation the question should be translated into a compliant ontology triple along with the suitable operators.

Question complexity can be categorized based on the number of variables and triples depending on the number of concepts mentioned in the question and the structure of the ontology. Question processing was approached based on several techniques; (i) interrogative question header, (ii) controlled natural language, (iii) focus and type of answer, and (iv) number of triples ; where important expressions that carry hints for the answer is detected and then translated into formal query format. Besides understanding the question, another challenge remains, which is to map the expressions in the questions with the ontology concepts.

Manual ontology mapping for SPARQL query rewriting has been studied in (Makris, Gioldasis, & Bikakis, 2010) by focusing on the mapping of formal specification. A similar approach for question to query mapping based on clarification dialogue that involves the user to disambiguate the correct resources to use was provided in (Lopez, Fernández, Motta, & Stieler, 2011). An enhancement was provided in (Damljanovic, Agatonovic, & Cunningham, 2011) by assigning confidence score for each of the ranked suggested disambiguated resource. However, it can be argued that these methods demand users to have some familiarity with the scheme and can lead to wrong result if generated by a novice user.

To the best of our knowledge semantic QA has only been addressed to process questions with low complexity; (i) simple questions where only one variable and one triple is involved, and (ii) constraint-based expressions, where an aggregation operation and only one triple is generated from the questions. In this paper we focus on the computational method for question with constraints. The first part of the paper introduces the study. The second part gives an overview of related works. The third part introduces the computational model for composite semantic QA. Part four presents the discussion and conclusion.

TRANSLATION OF NATURAL LANGUAGE QUESTION TO SPARQL QUERY

The semantic QA challenges can be described by three facets (Figure) namely *source heterogeneity*, answer formulation and question understanding. The source heterogeneity relates with the number of sources that need to be referred to answer the question. This is challenging because typically the structure of the ontology is unknown and the correct concept name to be included in the SPARQL query depends on the source’s structure.

The question understanding involves the focus identification, categorization and linguistic disambiguation. However, question understanding is also taxed because of the high question pattern variation, the depth and number of information required in the question and correct mapping between question and ontology concepts.

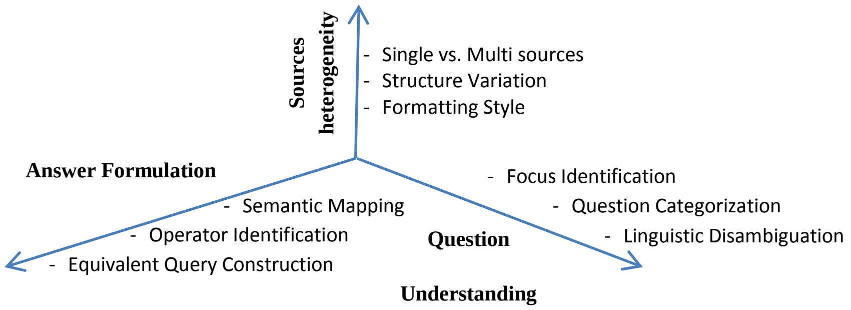

Figure 1. Conceptual Space of Semantic Question Answering Challenges

Formulating the answer for the question demands correct semantic mapping, operator identification and correct query translation. Most existing works in QA has focused on question understanding and several approaches are available. The question category was introduced by (Ferre & Hermann, 2011) and extended to include arithmetic and contextual category. Since this question category is not specifically designed for semantic answering, we refer to (Gao, Liu, Zhong, Chen, & Liu, 2011) for question depth categorization.

There are four question types namely *selection*, which indicates a straightforward question and simple answer construction. The *arithmetic* category is assigned for questions that involve some aggregation and modifier operation, such as performing answer sorting (ascending or descending) and mathematical operation (such as count and sum). The *path* category is assigned for composite questions; which means that the question can be broken into more than one atomic questions. The *path* type often has more than one variable to be manipulated. The *contextual* category defines question that contains fuzzy adjective, such as big and long which may not be matched easily from the ontology concepts and demand user definition if internal vocabulary that relates the adjective and concept property is not utilized.

However, most of the existing semantic QA works focus on selection typed questions (Damljanovic et al., 2011; Kaufmann, Bernstein, & Zumstein, 2006; Lopez, Uren, Motta, & Pasin, 2007). Only (Cimiano, Haase, & Mantel, 2007) has worked on constraints typed questions. However, this method is inefficient because accuracy of the translated query cannot be promised. Furthermore, questions that contain constraint expressions involve aggregation operators, which have not been explored in-depth previously. In the next topic we present a model for constraint-typed question processing.

COMPUTATIONAL MODEL FOR CONSTRAINT-TYPED SEMANTIC QUESTION ANSWERING

A complex NL question may require constraint and aggregate operations, which can be divided into *comparative*, *superlative*, and quantification. The basic approach in identifying constraints is by keyword-based or gazetteer approach, where a vocabulary of the relevant keywords are stored. This includes question identifier such as 'How many' and 'How much' which is tagged by <WHADP> or <JJ> to indicate a quantification expression, or arithmetic operation such as 'COUNT' and 'SUM'. The *comparative* expression is detected by the occurrence of the tagged part-of-speech pattern <JJR><IN> which parses fragments such as 'larger than', while the identification of *superlative* expression is by detecting the <JJS> tag. Superlative verb lexicons such as 'largest', 'smallest', 'greatest', and 'biggest' is stored to represent its orientation. For example, the orientation of 'largest', 'greatest' and 'biggest' is descending while for 'smallest' is ascending. The superlative verb lexicons are paired with the matching ontology property.

For example, in the question 'How many states does the Colorado river run through?', the question requires counting the number of results generated by executing a SPARQL query on states that have Colorado run through, using an object property called 'runThrough'. Another variable is introduced here namely *hasCount*. To construct the SPARQL, the *focus* is replaced with a variable, in this case '?vo'. Since the domain of 'runsThrough' is 'river' and 'state' is the range, the SPARQL is

```
PREFIX geo:<http://www.mooney.net/geo#>
SELECT DISTINCT (COUNT(*) AS ?number)
WHERE {geo:colorado2 geo:runsThrough ?v0 .}
```

We created a template to insert the namespace prefix indicate the target ontology at the beginning of the SPARQL query followed by the statement DISTINCT COUNT to instruct the reasoner to perform COUNT operation, in comformant with our initial question.

Besides questions that contain 'COUNT' expression, we also process questions with superlative expression. For example, in the question 'What is the state with the lowest population?', the detected concepts are 'state' as class and 'population' as data type property with the matching property namespace 'statePopulation'. We have also detected 'lowest' as the indicator of numeric expression with ascending operation as the orientation of the operation. In this example there is no mention of instance and only one variable and one triple are needed in the SPARQL query. In this question the SPARQL query is

```
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX geo:<http://www.mooney.net/geo#>
SELECT * WHERE {?v0 geo:statePopulation ?v1 .}
ORDER BY ASC(xsd:float(?v1)) LIMIT 1
```

A SPARQL query can be produced according to several styles, as shown in Figure 2. In this question the linguistic triple is <*river*, *?*, *texas*>while the *focus* is *river*. Note in this question the predicate is unknown, so the translator loads the properties shared by the *focus* and the c_o. The returned property is *runsThrough*. However, in this question another property is needed namely *length*, which will be used to supply information so that the 'longest river' can be computed. In the scenario of the geography dataset however, this challenge is minimum since there is only one data type property connected to the class *river*. Alternatively controlled vocabulary and consolidation dialogue can be utilised to disambiguate the property.

```
PREFIX geo:<http://www.mooney.net/geo#>
PREFIX xsd:
<http://www.w3.org/2001/XMLSchema#>
SELECT ?v0 WHERE {
        ?v0 geo:runsThrough geo:texas .
        ?v0 geo:length ?v1.}
ORDER BY DESC(xsd:float(?v1)) LIMIT 1;
```

```
PREFIX geo:<http://www.mooney.net/geo#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
SELECT ?v2 WHERE {
                ?v0 geo:runsThrough geo:texas .
                ?v0 geo:length ?v1.
OPTIONAL {
                ?v2 geo:runsThrough geo:texas.
                ?v2 geo:length ?v3.
FILTER ( xsd:float(?v3) >xsd:float(?v1))}}
```

Figure 2. Possible SPARQL queries for 'What is the longest river in texas?'

Our proposed computational model for answering semantic questions with constraints as shown in Figure is comprised of three processes namely linguistic processing, semantic mapping and SPARQL construction. The emphasis of the model is the linguistic processing which is separated into two tasks called explicit and implicit linguistic processing. The explicit linguistic processing identifies variables like question focus, answer type, concept annotation and linguistic triple identification. The implicit linguistic triple focusing enriched the information detected earlier by refining the linguistic triple, constraints expressions identification, superlative expression and constraints operation identification, extracting the concepts that are connected to the question focus and identifying the connective variable. This information is then used to disambiguate the mapping between the NL terms and ontology concepts. Then the SPARQL query is constructed. The composite semantic QA processing can also be seen as a layered processing technique because results from each of the atomic question is passed to the next atomic question., compared to federated question processing which runs the processing in parallel and finally combining the results.

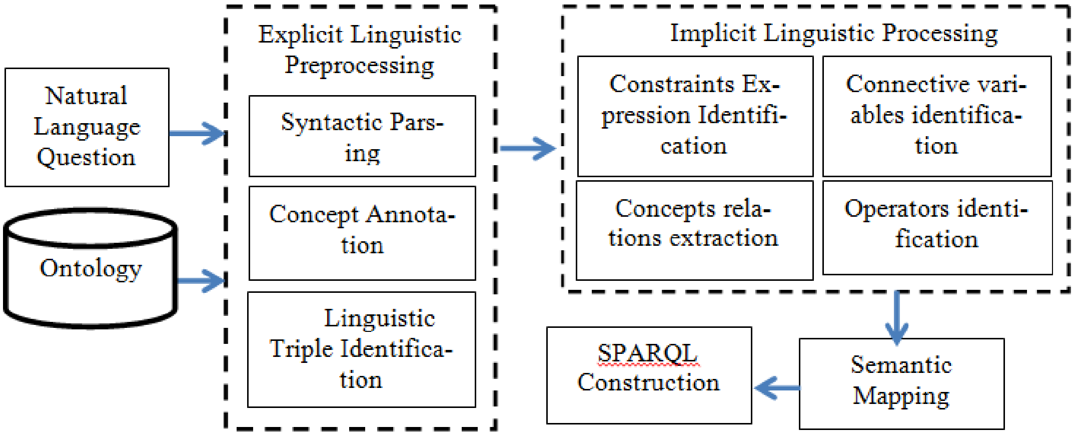
Natural Language Question
Ontology
Explicit Linguistic Preprocessing
Syntactic Parsing
Concept Annotation
Linguistic Triple Identification
Implicit Linguistic Processing
Constraints Expression Identification
Connective variables identification
Concepts relations extraction
Operators identification
Semantic Mapping
SPARQL Construction

Figure 3. Computational Model for Answering Semantic Questions with Constraints

The implemented model was evaluated in terms of variables extraction correctness where out of 84 constrained-typed questions in the geography dataset[1] used, the linguistic triple has 74 correct, the constraint expressions and operator have 100% correct identification and the SPARQL had 34 correct constructions.

DISCUSSION AND CONCLUSION

Semantic QA has high potential because there is increasing amount of semantic knowledge available in the web due to advancement of Web 3.0. However, the semantic knowledge is in structured form and requires formal language (i.e, SPARQL) before the answer is obtained. This indicates a necessity to construct the query on behalf of the user without mastering the structure of the ontology and proficiency in the formal query.

This paper focuses on NL question translation into SPARQL because NL in the most convenient way to pose questions compared to other methods such as form based which has scope limitation, keyword-based input which could not express the complete human thinking and graphical-based querying which might be confusing. This paper has presented a computational model for constraints-typed questions which has limited exploration due to its complexity.

For future work we will be working on the refinement of the technique and testing the work with larger dataset. We will also investigate the effect of domain specific application

[1] Taken from the Mooney's Geography NL query https://files.ifi.uzh.ch/ddis/oldweb/ddis/research/talking-to-the-semantic-web/owl-test-data/index.html

versus heterogeneous and linked data connectivity. Developing alternative methods would also be interesting.

ACKNOWLEDGMENTS

This paper is one of the activities in a project entitled "Logical Semantic Structured Language Model for Knowledge-Based Society" (LRGS/TD/2011/UITM/ICT/03) sponsored by the Ministry of Education, Malaysia. I therefore thank the sponsor for the encouragement and cooperation on this work.

REFERENCES

Cimiano, P., Haase, P., & Mantel, M. (2007). *ORAKEL: A Portable Natural Language Interface to Knowledge Bases Introduction* (p. 78).

Damljanovic, D., Agatonovic, M., & Cunningham, H. (2011). FREyA: an Interactive Way of Querying Linked Data Using Natural Language. *Proceedings of the 8th international conference on The Semantic Web* (pp. 125–138).

Damljanovic, D., Agatonovic, M., Cunningham, H., Court, R., & Street, P. (2010). Identification of the Question Focus: Combining Syntactic Analysis and Ontology-based Lookup through the User Interaction. *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2010)*.

Ferre, S., & Hermann, A. (2011). Semantic Search: Reconciling Expressive Querying and Exploratory Search. *ISWC'11 Proceedings of the 10th International Conference on The Semantic Web* (pp. 177–192).

Gao, M., Liu, J., Zhong, N., Chen, F., & Liu, C. (2011). Semantic Mapping From Natural Language Questions To Owl Queries. *Computational Intelligence*, *27*(2), 280–314.

Kaufmann, E., & Bernstein, A. (2007). How useful are natural language interfaces to the semantic web for casual end-users? *The Semantic Web, 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference* (pp. 281–294).

Kaufmann, E., Bernstein, A., & Zumstein, R. (2006). Querix: A Natural Language Interface to Query Ontologies Based on Clarification Dialogs. *Proceedings of the 5th International Semantic Web Conference (ISWC 2006)* (pp. 980–981).

Lopez, V., Fernández, M., Motta, E., & Stieler, N. (2011). PowerAqua: supporting users in querying and exploring the Semantic Web. *Semantic Web-Interoperability, Usability, Applicability*.

Lopez, V., Uren, V., Motta, E., & Pasin, M. (2007). AquaLog: An ontology-driven question answering system for organizational semantic intranets. *Web Semantics: Science, Services and Agents on the World Wide Web*, *5*(2), 72–105. doi:10.1016/j.websem.2007.03.003

Makris, K., Gioldasis, N., & Bikakis, N. (2010). Ontology Mapping and SPARQL Rewriting for Querying Federated RDF Data Sources. *Proceedings of the 2010 international conference on On the move to meaningful internet systems: Part II* (Vol. 31, pp. 1108–1117).

Sharef, N. M., & Mohd Noah, S. A. (2013). Natural Language Query Translation for Semantic Search. *Journal of Digital Content, Technology and Applications*, (in press).

Sharef, N. M., & Mohd, S. A. (2012). Soft Queries Processing In Natural Language. *International Conference on Ubiquitous Information Management and Communication* (pp. 1460–1466).