

Open API for Ontology Building based on Sentence Triplet Extraction

Il-Chul Moon, Se-Jin Nam, Dong-hyun Choi, Key-Sun Choi

Semantic Web Research Center, Computer Science Department, KAIST
335 Gwahangno Yuseong-gu, Daejeon, Republic of Korea, 305-701
{imoon, sjnam, cdh4696, kschoi}@world.kaist.ac.kr

Abstract. Ontology is one of the important semantic web enablers. However, there are few open Application Programming Interfaces, or APIs, to support developers in terms of building ontology. Often, the developers are asked to use a specific tool that does not open its API in a developer friendly manner. Thus, we provide a set of open APIs for ontology building. These APIs aim to help ontology building through sentence triplet extractions. Particularly, the APIs are used by a CoreOnto workbench semi-automatically generating an ontology structure from a Wikipedia corpus. Because of the open APIs, we expect that the ontology engineers can easily access this APIs through the internet and obtain the same ontology that might be generated by a sophisticated stand-alone ontology workbench. These open APIs are the first step to make this ontology and semantic web community grow through an open infrastructure.

Keywords: Open API, ontology technology, ontology building, semantic web

1 Introduction

Ontology is a core technology of semantic web [1]. Ontology enables transferring complex knowledge base from one system to another [2]. Also, ontology provides a data representation format to be used for various semantic web services [3]. If we personalize ontology for each individual, we even can represent the individual's personal behavior pattern, surrounding environment, involved social relations, etc. These are important technical advances allowed by the ontology.

One of the important ontology technologies is the ontology building technology, such as automatic identification of ontology taxonomy, ontology concepts, relations, etc. Traditionally, the ontology building largely depends on the human annotation efforts supported by a software workbench, such as protégé [4]. Recently, the ontology building technology is implemented by automatic tools, such as Text2Onto [5]. However, to our knowledge, there is no open API that extracts a triplet structure [8] from a document sentence and expands an ontology structure. Therefore, we provide such an innovative ontology building algorithm through our open API system.

This paper introduces open APIs and an application workbench utilizing the open API. We expect that our proposed APIs adapted to other systems for ontology building.

Business value proposition and solution position. We propose our system to service a sophisticated sentence triplet extraction for ontology building. Since the

triplet extraction required the named entity recognition, the sentence parsing, the ontology class matching technologies, a single developer cannot implement this triplet based ontology building easily. We provide open APIs that service the above technologies freely. Our implemented innovative idea, triplet based ontology extraction, is in the process of patent filing.

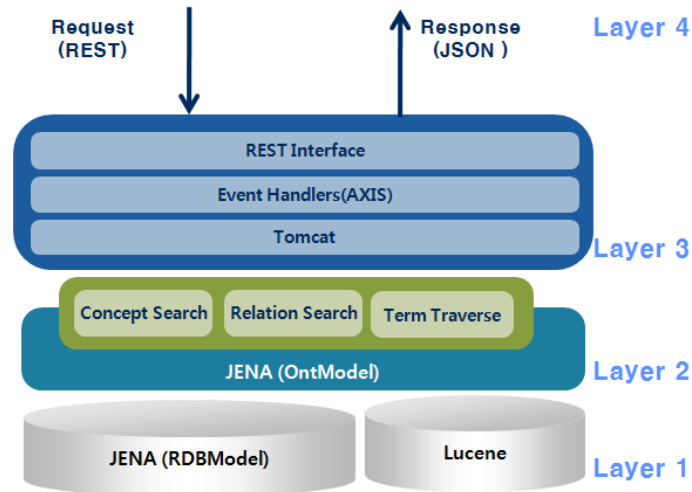


Fig. 1. The system architecture of our open API. The request and response are based on HTTP request and response.

3 System description

Our system is the open API for ontology building. This system can be described from the technical perspective and the theoretical perspective. The technical perspective is the brief description about how we enabled the open API system that is easily accessible to developers. The theoretical perspective is the actual input and output to the system as well as the process of the triplet extraction. We first explain the technical aspects and later show the theoretical aspects.

3.1 Technical aspects of the system

Figure 1 is the system architecture of our open API system at high level. The API should be accessible to semantic web and ontology developers around the world. Thus, we enabled the access to our API by using HTTP. We implemented the system by using the most up-to-date supporting tools, such as JENA [6]. From the input viewpoint, the system accepts input in the REpresentational State Transfer, or REST, manner. Particularly, we use the simple GET method of HTTP requests. The following is the example input to the system. From the output viewpoint, the system generates an OWL text output [7] corresponding to the extracted triplet. We provide this ontology building approach through this open APIs. Table 1 briefly lists the APIs.

Table 1. Ontology building APIs

API Definition	API Query type	Required parameters	Response type
Triplet extraction from a sentence	getTriple	Text Sentence	RDF/OWL
Triplet mapping to an existing ontology ¹	getTripleList	First, second and third arguments of the triplet to search, List order (ascending/descending), Number of returned triplet, Search offset	Test/JSON

3.2 Theoretical aspects of the system

Our innovative idea for the ontology building is utilizing the triplet structure of a text sentence. As displayed in Figure 2, the users can provide a sentence implying a triplet, eventually a small ontology structure that can be mapped to an existing ontology. From the raw text to the small ontology, we need a set of natural language processing techniques: 1) named entity recognition, 2) sentence parsing, and 3) triplet identification. Through our open API, we provide the above techniques freely, so that the users can only focus on the corpus management and the generated ontology mapping.

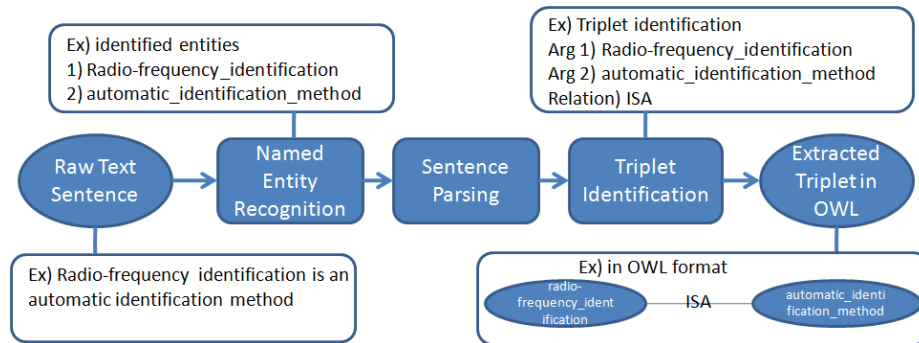


Fig. 2. The theoretic triplet extraction and OWL conversion process. This process is wrapped and available through an open API in our system

4 Application

This section introduces an application example and the robustness of the open API system. Firstly, the application is a CoreOnto workbench which semi-automatically generates an ontology structure from a corpus with human experts' help. Secondly,

¹ We used the KAIST triplet collections. This triplet set can be obtained by a request to kschoi@world.kaist.ac.kr.

the robustness is examined to ensure that the open API can be robust enough to support the semantic web programmers devices over the internet.

4.1 CoreOnto: a ontology building workbench

CoreOnto is an application utilizing this open API. CoreOnto is a semi-automatic ontology building tool, and its automatic ontology building relies on the above triplet extraction. Figure 3 shows the main view of the CoreOnto workbench.

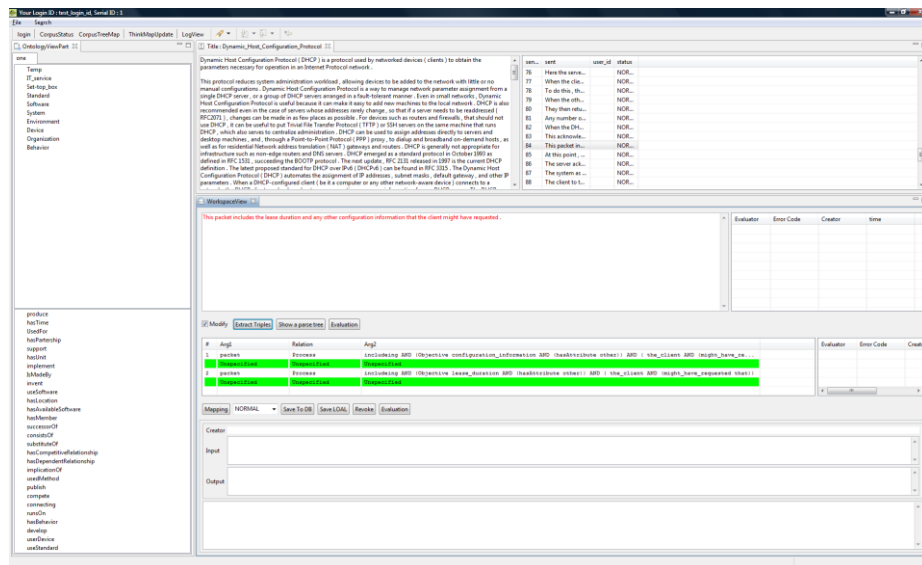


Fig. 3. CoreOnto workbench: a semi-automatic ontology building tool

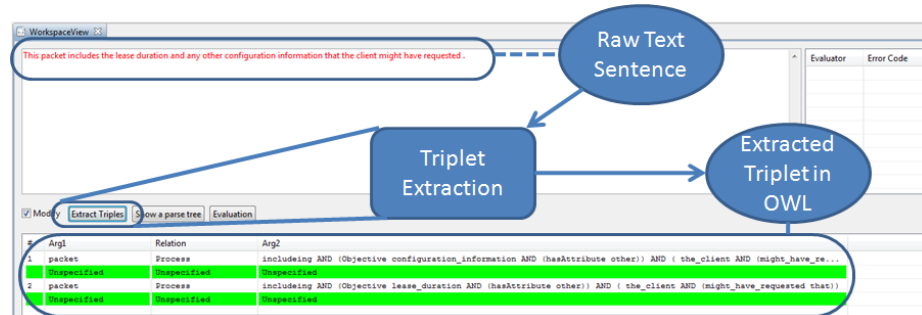


Fig. 4. A part of GUI that uses the triplet extraction open API in the CoreOnto workbench

Figure 4 is the specific workbench function that utilizes the triplet extraction open APIs. The sentence is provided by a Wikipedia crawler. Our triplet extraction API can identify multiple triplets if the sentence implies more than one. After extracting the

triplets, the relations and the arguments are matches to the existing ontology, and the matched triplets are attaches to the appropriate location of the ontology and expand it.

4.2 Robustness test of the open API

Considering the actual usage of our open APIs, a critical question is the robustness of the APIs. If the APIs are not robust enough, then the developers might better off by coding the above natural language techniques in their own machine. However, we demonstrate that our open APIs are robust enough to be served via the internet. Figure 5 shows that the triplet extraction API calls do not take more than 45 sec in an extreme environment test. Also, the calls are handled reasonably quickly compared to an ordinary ontology search API calls.

Table 2. API test system environment

System side	System specification category	Specification value	System side	System specification category	Specification value
Server	CPU	Intel Core2Duo 1.8GHz	Client	CPU	Intel Pentium 3.4GHz
	Memory	1 Gigabyte Ram		Memory	1 Gigabyte Ram
	OS	Window XP		OS	Window XP
	Web server	Tomcat V5.5 Axis2		Test program	OPENSTA 1.4

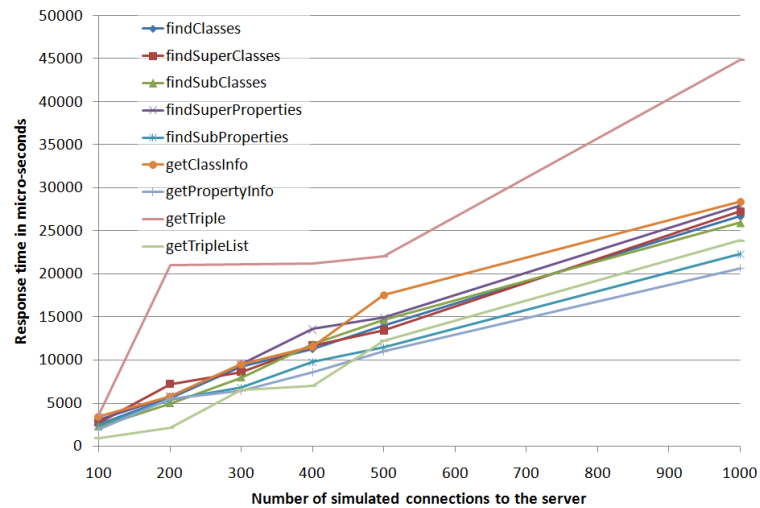


Fig. 5. Robustness test for the open API system. Each query functions are tested under various levels of server stress from 100 simulated connections to 1000. Except the triplet finding algorithm, the open API system respond within 30sec even under the most stressed situation.

5 Conclusion

This paper introduces an open API system for ontology building based on the triplet extraction approach. This is just a proof-of-concept for API service to support the semantic web development. Thus, we will expand this API system to include various functions frequently used by semantic web developers and ontology engineers. Meanwhile, we provided 1) an application of this open API and 2) the robustness test result of our APIs. This remote API 1) reduces the development time for developers, 2) facilitates the internet access to various semantic web resources, and 3) provides the test-bed for new theoretical idea in the semantic web and ontology domain, which is a fundamental reason why KAIST, an academic institute, provides this system.

Acknowledgements

This work was supported in part by Institute for Information Technology Advancement (IITA), Korea (A1100-0601-0102). Additional support was provided by SWRC – the Semantic Web Research Center (SWRC), KAIST. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of the IITA.

References

1. Davies, J., Fensel, D., and van Harmelen, F., ed. (2002) *Towards the Semantic Web - Ontology-Driven Knowledge Management*. Wiley.
2. Gruber, T. R. (1993) *A Translation Approach to Portable Ontology Specifications*. Knowledge Acquisition, Vol. 5.
3. McIlraith, S. A., Son, T. C., and Zeng, H. (2001) *Semantic Web Services*, IEEE Intelligent Systems, Vol. 16, Num. 2, pp. 46-53.
4. Noy, N. F., M. Crubezy, et al. (2003) *Protege-2000: an open-source ontology-development and knowledgeacquisition environment*. AMIA Annual Symposium Proceedings, pp. 953
5. Philipp, C. and Volker, J. (2005) *Text2Onto - A Framework for Ontology Learning and Data-driven Change Discovery*, In Proceedings of the 10th International Conference on Applications of Natural Language to Information Systems - NLDB'05 (pp. 227-238), June 15-17
6. Carroll, J. J., Dickinson, I., Dollin, C., Reynolds, D., Seaborne, A., and Wilkinson, K. (2004) *Jena: Implementing the Semantic Web Recommendations*, In the proceeding of 13th World Wide Web Conference, pp 74-83
7. McGuinness, D. L. and van Harmelen, F. (2003) *Owl web ontology language overview*, <http://www.w3.org/TR/2003/WD-owl-features-20030331/>, March 2003
8. Choi, D. H. and Choi, K. S. (2008) *Automatic Triplet Relation Extraction by Dependency Parse Tree Traversing*, EKAW 2008 Poster, Catania, ItalySmith, T.F., Waterman, M.S.: Identification of Common Molecular Subsequences. J. Mol. Biol. 147, 195--197 (1981)