IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Special Issue, 4, Feb 2019 E-ISSN: 2347-2693

Semantic Ontology Extraction in Heterogeneous Text Documents

T. Deepaalakshmi

Department of Computer Applications, Bharathidasan University, Thiruchirappalli, India

Corresponding Author: deepaarassu88@gmail.com

Available online at: www.ijcseonline.org

***Abstract*—**Ontology Extraction is an important role in the Semantic Web as well as in knowledge management. The emergence of Semantic Web and the associated technologies promise to make the Web a meaningful experience. On the contrary, success of Semantic Web and its applications depends largely on utilization and interoperability of well-formulated ontology bases in an automated heterogeneous environment. Ontology is what exists in a domain also how they relate with each other. The advantage of ontology is that it represents real world information in a manner that is machine understandable. This leads to a diversity of interesting applications for the benefit of the target user groups. Ontology defines the terms used to describe and represent an area of knowledge. Ontologies are significant for applications that need to search across or merge information from diverse communities. In this paper, we present our move toward to extract relevant ontology concepts and their relationships from a knowledge base of heterogeneous text documents.

***Keywords*—**heterogeneous, knowledge, machine understandable, Ontology Extraction, Semantic Web.

I. INTRODUCTION

The Semantic Web is a major research initiative of the World Wide Web Consortium (W3C) [1] to create a metadata-rich Web of resources that can describe themselves not only by how they should be displayed (HTML) or syntactically (XML), but also by the meaning of the metadata. We believe Semantic Web as next generation Web that provides great benefits in Web Services, Internet Commerce, and further promising application areas. Though, Semantic Web is still in its primary stage means not fully implemented and has lots of unsolved problems. One of the most important problems is to extract data from heterogeneous documents in such way that it has to recognize by machine, which we call ontology extraction.

A basic approach for ontology extraction is inmanual. Most of the present research focuses on exploiting various methods to generate ontology automatically or semi-automatically. Manual ontology building is a time taking action that requires a lot of efforts for knowledge domain acquisition and knowledge domain modeling. In order to overcome these problems various methods have been developed, including systems as well as tools that automatically or semi-automatically, by means of text mining and machine learning techniques, lets to generate ontologies. The research fields which study this issues is usually called “ontology generation”or“ontology extraction” or “ontology learning”. However, most approaches have “only” considered one step in the overall ontology engineering process [2], forexample, generating concepts & relationships[3] or extracting concepts & relationship whereas one must consider the overall process when building real-world applications. In this paper, we express our approach for ontology extraction from an existing knowledge base of heterogeneous documents. We need Information Extraction from heterogeneous text because it gives direct access to knowledge when in textual format only relevant information is accessed by people KnowledgeSharing.

II. RELATEDWORKS

Two main approaches have been developed in ontology mining. The first one facilitates manual ontology engineering by providing natural language processing languages, and ontology import tools. The second method is based on machine learning and automated language processing techniques to extract concepts and ontological relations from structured and unstructured data such as databases and texts. A number of systems have been projected for ontology extraction from text. We express some of them in the following.

ASIUM [4] extracts verb frames and taxonomic knowledge, stands on statistical analysis of syntactic parsing of texts. Text-To-Onto [5] is an Open source ontology management infrastructure, by means of a tool suite for building ontologies from initial core ontology. It merges knowledge acquisition and machine learning techniques to discover conceptualstructures.

Information Retrieval [6] is a domain independent that creates clusters of the words appearing in the text. The scope of this is to construct a hierarchy of concepts. Its learning technique is based on distributional approach: nouns playing the same syntactic role in sentences with the same verb are grouped together in the sameclass.

Effective ontology management in virtual learning environments[7] is a semi-automatic data driven topic ontology which integrates machine learning and text mining algorithms. Major features are represented by automatic keyword extraction from documents given as an input to the system (the extracted keywords are "candidate concepts" of the ontology) and by the concepts suggestions generation.

III. APPROACH FORONTOLOGY EXTRACTION

Ontology is a basic building block for semantic web[8]. A dynamic line of research in semantic web is focused on how to build and evolve ontologies using the information from different ontological sources like txt, doc, ppt, pdfetc inherent in the domain. A huge part of the IT industry uses software engineering methodologies to build software solutions that solve real-world problems. Ontology Building procedure consists of followingphases.

A. *Clustering*

We have implemented statistical [9] and data mining algorithm [10] in order to identify the concepts and their relationship in the resulting ontology. This technique aims to build ontologies using a data mining approach called cluster mining from domain repositories written inXML.

Algorithm: Generating Concepts and relations.
Input: Folder holding heterogeneous file
Output: Dynamically produced XML data by parsing the contents of files from ontology testingfolder.

❖ *Begin*

Step1: Read the entire file names from input folder. Step2: Create a string buffer variable to collect all the file names. Step3: Generate a temporary string buffer to read content of each file. Step4: Process all data of file based on end of sentence.
Step3: Create a temporary string buffer to read contentof each file.
Step4: Process all data of file based on end of sentence.
Step5: Using short-term string buffer which will list the number of possibilities of meaningless words in sentence, Cluster the data by filtering it from meaningless string content.
Step6: Mark first word of sentence as parent and next beginning word will be marked as child. Step7: Continue to read the entire sentences from the folder.

❖ Stop

B. *Harmonization*

This is an optional step that is needed when the user wants to "harmonize" the extracted ontology with the available knowledge bases.

With the term ontology harmonization, we wish to refer to the ability of harmonizing two or more ontologies in a unique ontology in order to improve the available knowledge base. It is severely related to two main issues: ontology matching [11] for the recognition of correspondences between ontologies and ontology merging [12] for the actual fusion of those ontologies. Major aim of harmonization is extracting concepts and relations means for input string it has to display list of all the match able relations from the inputstring.

Algorithm: Extracting concepts and relations
Input: Testing string query
Output: Displaying list of the entire the match able relations from the input string.

❖ □□Begin

Step1: Read the Input text.
Step2: Compare input test string by the concept from ontologydata.
Step3: Search input text with group of relations from ontologydata.
Step4: Read the number of term frequency of the input string appearing in the ontology data.
Step5: Display the number of strings emerging both as concept and relation.

❖ *Stop*

IV. RESULTS

This chapter talks about the results attained from the developed system, mainly it shows xtracted ontology data, constructed "concept & relationship" data created using the ontology data & verified ontology data process. In fig1 we can see ontology server containing a variety of options. Here the first process you have to browse the folder which contains various heterogeneous documents. In the below figure shown we are browsing the folder from "c:\Users\Kiran\Desktop\Test" location.Our experimentation has been made considering the TXT, DOC andPDF formatsconsequently our Test folder contains three different format files. The fig.2 indicates that.

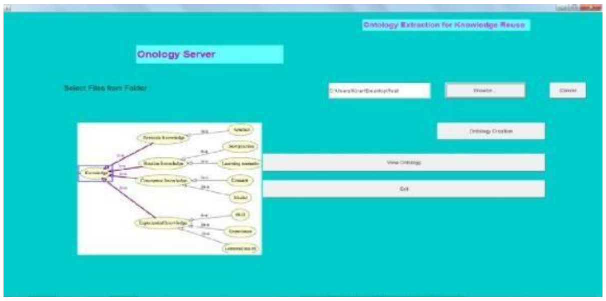
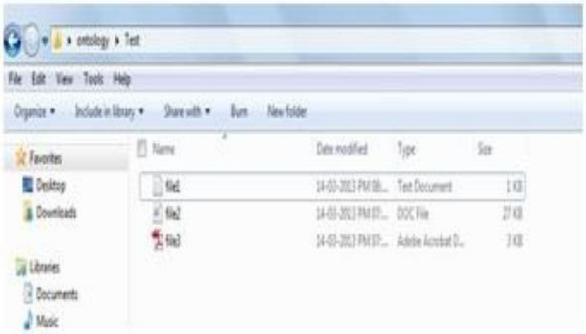


Fig.1: ontology server containing various options.

File1 is of text format that contains. “Hypertext Markup Language, the languages of the World Wide Web, allows users to generates Web pages that contain text, graphics and pointer to other Web pages.HTML provides tags to build the document lookattractive”


ontology ▸ Test
File Edit View Tools Help
Organize ▾ Include in library ▾ Share with ▾ Burn New folder
Favorites
Desktop
Downloads
Libraries
Documents
Music

Name	Date modified	Type	Size
file1	14-03-2013 PM 08...	Text Document	1 KB
file2	14-03-2013 PM 07...	DOC File	27 KB
file3	14-03-2013 PM 07...	Adobe Acrobat D...	3 KB

Fig.2:Test olderwhichcontainsthreedifferentformatfiles

File2 is of doc format, that contains “A HTML document is minute and hence easy to send over the net. It is minute because it does not contain formatted information.”

File3 is of pdfformat, that contains “HTML is platform self-directed. HTML tags are not case- sensitive.”

We want to analyze our input data so in all files we taken small amount of text. Our system works well with huge amount of data also. Subsequent to browsing the input folder, to build the ontology data we have to click ontology creation tab. As soon as you click the ontology formation tab within few seconds our system will generate ontology data. Subsequent to generating ontology data it shows “ontology creation has been successfully completed.” which is shown in fig.3.

Once the ontology creation successfully completed you can see ontology data by clicking view ontology tab. When you click view ontology label it displays two xml files. In our system ontology data is accumulated in xml format. First xml file holds ontologydata.

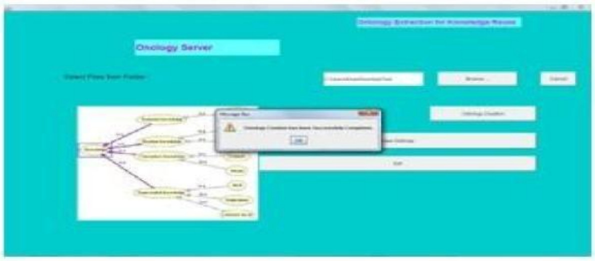
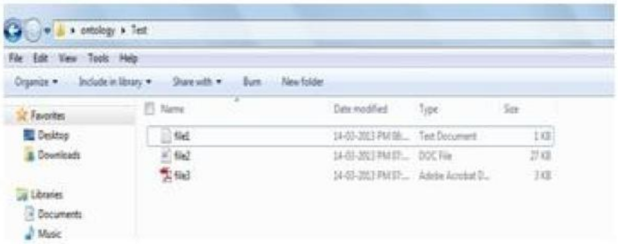


Fig.3: Ontology sever displaying message after creating ontology data

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<OntologyString>
<Text>
<Text><Hypertext Markup Language, languages World Wide Web, users produces web pages include text, graphics pointer Web pages></Text>
<Text><HTML tags make document attractive></Text>
<Text><Text><html document small easy send net></Text>
<Small Include Formatted information></Text>
<Text><HTML Platform IndePendent></Text>
<Text><HTML tags Case Sensitive></Text>
</Text>
</OntologyString>
```


ontology ▸ Test
File Edit View Tools Help
Organize ▾ Include in library ▾ Share with ▾ Burn New folder
Favorites
Desktop
Downloads
Libraries
Documents
Music

Name	Date modified	Type	Size
file1	14-03-2013 PM 08...	Text Document	1 KB
file2	14-03-2013 PM 07...	DOC File	27 KB
file3	14-03-2013 PM 07...	Adobe Acrobat D...	3 KB

Fig.4: Contents of first xml file

File2 is of doc format, that contains “A HTML document is minute and hence easy to send over the net. It is minute because it does not contain formatted information.”

File3 is of pdfformat, that contains “HTML is platform self-directed. HTML tags are not case- sensitive.”

We want to analyze our input data so in all files we taken small amount of text. Our system works well with huge amount of data also. Subsequent to browsing the input folder, to build the ontology data we have to click ontology creation tab. As soon as you click the ontology formation tab within few seconds our system will generate ontology data. Subsequent to generating ontology data it shows “ontology creation has been successfully completed.” which is shown in fig.3.

Once the ontology creation successfully completed you can see ontology data by clicking view ontology tab. When you

click view ontology label it displays two xml files. In our system ontology data is accumulated in xml format. First xml file holds ontologydata.

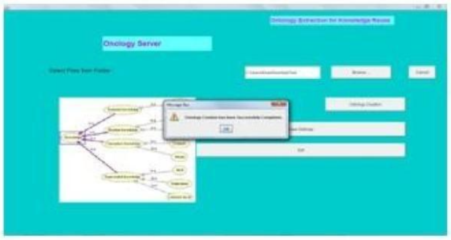


Fig.3: Ontology sever displaying message after creating ontology data

Fig.4: Contents of first xml file

The fig.4 shows content of first xml file. Here you can examine that our system concatenated all contents of different files. Then content divided bysentence.

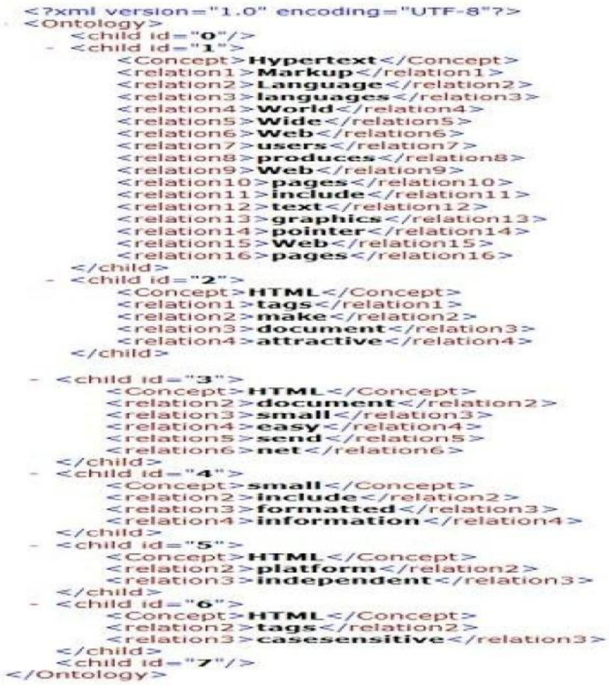
For example by considering content of file1, I will give details of the working of our system. First sentence of file1 is stored similar to below.

"Hypertext Markup Language, the languages of the World Wide Web, allows users to create Web pages that include text, graphics also pointer to other Web pages."

Second sentence of file1 is stored similar to below. "HTML provides tags to make the document lookattractive". Lastly our system removes stop words (unrelated words) from every sentence. Unrelated words means in first sentence the, of, allows, to, that & other words be having no importance when creating ontology data. So those words have been trimmed beginning the sentence. So trimmed content with admiration to first sentence of file 1 is "Hypertext Markup Language, languages World Wide Web, users produces web pages comprise text, graphics pointer Web pages" Trimmed content with respect to second sentence of file 1 is "HTML tags make document attractive" Similar process applied to whole content & stop words have removed from each sentence (refer fig.4 for output).Next the each sentence of ontology data is stored in "Concept-Relationship" manner which is useful when extracting ontology data. In every sentence first word is stored as concept & next words will be stored as relations.

Once the ontology data is created next optional step is to check match able relations from sentence for input string. In our system it is working well. Suppose for instance your searching html as input string then it will display machable relations. Math able relations for html are tags, make, document, attractive, minute, easy, send, net, platform, self-governing, case sensitive. It also shows in which file the particular sentence is found. So you can simply find the exact information.

In fig.5 each sentence first word is stored as concept & next words will be stored as relations. It does not signify that you have to search only concept. You can search every word means the particular input string is treated as concept related words are treated as relations. Suppose if you given input that is not there in documents then it will display the message "search not found, attempt with anotherconcept."



```
<?xml version="1.0" encoding="UTF-8"?>
<Ontology>
    <child id="0"/>
  - <child id="1">
        <Concept>Hypertext</Concept>
        <relation1>Markup</relation1>
        <relation2>Language</relation2>
        <relation3>languages</relation3>
        <relation4>World</relation4>
        <relation5>Wide</relation5>
        <relation6>Web</relation6>
        <relation7>users</relation7>
        <relation8>produces</relation8>
        <relation9>Web</relation9>
        <relation10>pages</relation10>
        <relation11>include</relation11>
        <relation12>text</relation12>
        <relation13>graphics</relation13>
        <relation14>pointer</relation14>
        <relation15>Web</relation15>
        <relation16>pages</relation16>
    </child>
  - <child id="2">
        <Concept>HTML</Concept>
        <relation1>tags</relation1>
        <relation2>make</relation2>
        <relation3>document</relation3>
        <relation4>attractive</relation4>
    </child>

 - <child id="3">
       <Concept>HTML</Concept>
       <relation2>document</relation2>
       <relation3>small</relation3>
       <relation4>easy</relation4>
       <relation5>send</relation5>
       <relation6>net</relation6>
   </child>
 - <child id="4">
       <Concept>small</Concept>
       <relation2>include</relation2>
       <relation3>formatted</relation3>
       <relation4>information</relation4>
   </child>
 - <child id="5">
       <Concept>HTML</Concept>
       <relation2>platform</relation2>
       <relation3>independent</relation3>
   </child>
 - <child id="6">
       <Concept>HTML</Concept>
       <relation2>tags</relation2>
       <relation3>casesensitive</relation3>
   </child>
   <child id="7"/>
</Ontology>
```

Fig.5: second xml file storing ontology data in concept-relationship manner.

V. CONCLUSION

In this paper, we have presented an ontology information extraction system to extract ontologies from a knowledge base of heterogeneous text documents. We have projected our approach to build the Concept and Relationship from heterogeneous documents which gives dynamically created XML data by parsing the contents of files. In our task harmonization is an optional step but it is needed to check whether build ontology is efficient or not, so we even projected our approach to extracting concepts and relations. Means when you give input as string query our system gives output as list of all the match able relations from the input string. Our work principally explains the ontology extraction process is general and is not domain dependent.

Thus ontology has been served as a mainly effective technique to solve semantic issues irrespective of anydomain.

REFERENCES

[1] M. Dean and G. Schreiber, "OWL Web ontology language reference," W3C Recommendation, Feb.2004.

[2] J. Euzenatand P. Shvaiko, Ontology Matching. Heidelberg, Germany: Springer-Verlag,2007.

[3] R. Farkas, V. Vincze, I. Nagy, R. Ormándi, G. Szarvas, and A. Almási, "Web-based lemmatisation of named entities," in Proc. TSD,vol.5246,LectureNotesinComputerScience,P.Sojka, A. Horák, I. Kopecˇek, and K. Pala, Eds. Berlin, Germany:Springer-Verlag, 2008, pp. 53–60.

[4] D.Faure and T. Poibeau, "First experiences of using semantic knowl- edge learned by ASIUM for information extraction task using INTEX," in Proc. ECAI Workshop Ontology Learning, vol. 31, CEUR Work- shop Proceedings, S. Staab, A.Maedche, Nédellec, and P. Wiemer- Hastings, Eds.,2000.

[5] A.Maedche and S. Staab, "The Text-To-Onto ontology learning environ- ment," in Proc. 8th Int. Conf. Conceptual Struct., Darmstadt, Germany,2000, pp.14-18.

[6] W.BFrakes and R. A. Baeza-Yates, Eds., Information Retrieval: Data Structures & Algorithms. Englewood Cliffs, NJ:Prentice-Hall,1992.

[7] M. Gaeta, F. Orciuoli, S. Paolozzi, and P. Ritrovato, "Effective ontology management in virtual learning environments," Int. J.Internet Enterprise Manage., vol. 6, no. 2, pp. 96–123,2009.

[8] A.D. Maedche, Ontology Learning for theemanticWeb.Norwell, MA: Kluwer,2002.

[9] C. D. Manning and H. Schtze, Foundations of Statistical Natural Language Processing. Cambridge, MA: MIT Press, Jun.1999.

[10] D. L. McGuinness, R. Fikes, J. Rice, and S. Wilder, "The Chimaera ontology environment," in Proc. AAAI/IAAI, 2000, pp. 1123–1124.

[11] R. Navigli and P. Velardi, "Semantic interpretation of terminological strings," in Proc. 6th Int. Conf. TKE, 2002, pp. 95–100.

[12] R. Navigli, P. Velardi, and A. Gangemi, "Ontology learning and its application to automated terminology translation," IEEE Intell. Syst., vol. 18, no. 1, pp. 22–31, Jan.2003.