International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 7, July - 2014

Ontology Development from Text Document for Search Engine

Trupti Hake.
Department of Computer Engineering,
SKNCOE,
Pune, India

Prof. Vaishali S. Deshmukh.
Department of Computer Engineering,
SKNCOE,
Pune, India

***Abstract*— In today's world we can get any type of information from Internet. Many web applications use Semantic webs. Ontology are elementary part of semantic webs. Also maintain static database for different e learning system is a challenging task. In case unstructured databases which contains the text files , it becomes difficult to retrieving of information and hence it leads failure in acquire accuracy of knowledge. Today's era is all about maximum knowledge gain in less effort. So the search system has to more prominent. so changing the structure of storing the data. Need and concept of semantic web has given birth to Ontology. Here we are representing a semi-automatic structure to build ontology from text document. And to retrieve information from ontology database Apache Lucene search engine which is developed.**

Keywords— Ontology; Semantic web; Knowledge database; Concepts.

I. INTRODUCTION

Today's era is all about e-learning system and all supporting knowledge databases. In any domain we have adopted this. But maintaining huge unstructured database and acquiring knowledge from it difficult task. so semantic web are used . Need of semantic web has done in evolution of concept called Ontology. A data model called Ontology can be used for more specific search and information gain in systems like knowledge base, e- learning, here aim of paper is building of Ontology from text document and changing the unstructured database to structure database and which lead to maximum information gain on search. Here we are representing a semi-automatic structure to build ontology from text document.

Ontologies attempt to represent or model human knowledge about the world. Ontology is used as data model in data warehouse, for data mining, visualization. Semantic web is all about manipulation of information so there should be a proper relationship between the objects. This implies use of ontology in semantic web application in large scale. And all this information is available in the text format. So here presenting a semi automatic ontology builder approach to build ontology from text document.

Ontologies are frequently used in the context of software and technology engineering. These can be grouped into two main categories, depending on whether they are used to describe the Knowledge of a domain (domain ontologies) or whether they are used as software artifacts in Software development processes. Here, we follow the ontology definition provided by Gruber (1993): 'an explicit specification of a conceptualization'. [1]

Ontologies are seen in Artificial Intelligence as fulfilling the need to capture knowledge in a manner which would make possible sharing and re-use and because of this they have become a key research topic in Artificial Intelligence in the last decade or so. In many respects, ontologies are building on the work in 'knowledge representation' conducted in AI over many years. However, their importance has extended to other domains such as electronic commerce, intelligent information integration, information retrieval and, as we have mentioned, Knowledge Management. [11]

II. LITERATURE SURVEY

Maizatul Akmar Ismail and et al suggested a dynamic ontology editor which will help academic researcher in different ways. They stated that academic research almost without exception involves literature searches. We know that there is huge amount of data information available so we can sort this using ontology and finally this will lend a hand for system to extract significant literature to researcher from ontologically based system. Researcher can find out relevant information on few plain queues such as paper based title and hypothesis. In this paper they introduce an ontology-based digital library system and detail the first component of the system that allows real time flexible ontology management. The flexibility aspect includes the modification of the ontology at any time, as well as full real time graphical representation and editing. We introduce our novel semi-formal representation scheme for ontologies that promotes ontology modifiability and adaptability. [2]

According to study of Matteo Gaeta and Stefano Paolozzi,(2011) ontologies have been frequently employed in order to solve problems derived from the management of shared distributed knowledge and the efficient integration of information across different applications. However, the process of ontology building is still a lengthy and error-prone task. Therefore, a number of research studies to (semi-)automatically build ontologies from existing documents have been developed. In this paper, we present our approach to extract relevant ontology concepts and their relationships from a knowledge base of heterogeneous text documents. They also show the architecture of the implemented system and discuss the experiments in a real-world context. Finding (semi-

IJERT

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

)automatic algorithms to extract ontology concepts from existing knowledge bases represents an important activity. They have described approach for ontology extraction from an existing knowledge base of heterogeneous documents. [1]

As stated by, semantic web technology involves the structuring of information using metadata to be interpreted or rather utilized by software agents in order to sort or find information more accurately and precisely. Semantics is essentially the meaning of symbols as pertaining to other related symbols, descriptions and links. Improving the accuracy of the search requires a set of associated, standardized rules and terminologies in the form of ontologies. Ontologies may provide a taxonomy or classification, which will improve the accuracy of queries and retrieve a result based on semantics of the terms. [6]

Leyla Zhuhadar and et al presented an ontology-based document-driven memory in E-learning that used two ontologies: a generic ontology related to the domain of training in general and a domain-specific ontology to deal with the application at hand. The research was divided into three domains: knowledge engineering, pedagogical design and Semantic Web, and utilized Topic Maps in the representation in the learning memory. [3][10]

Conventional information systems are built on top of a relational database that requires its data model to be stable. This lack of adaptability is very restrictive for systems that manipulate evolving or heterogeneous knowledge. The authors of this paper faced failure at the time of the development of an information system which is very heterogeneous and dynamic, it is practically impossible to define a stable database schema ahead. The widely accepted alternatives to relational databases are semantic web ontologies. This paper tries to fill the gap by proposing a methodology for designing ontology-backed software applications that make the ontology possible to evolve while being exploited by one or more applications at the same time. [4]

Ayesha Ameen (2012) has stated that Ontology is an integral part of semantic web. Ontology can be design and create metadata elements required for developing semantic web applications. The evolution of semantic web has encouraged creation of ontologies in many domains. In this paper they have describe various steps involved in creation of university ontology. University otology can be applied to particular university. They have used protégé 4.0 alpha tool. And finally they have conclude the result with successful creation of ontology of University and it can be reused or integrated into any university to facilitate efficient access and retrieval of information that can be automatically processed by machine. [6]

Huei-Zhen Gu (2012) stated that construction of a knowledge base is the fundamental of any computation knowledge sharing project. This research is aimed to explore a methodology of building an ontology-based knowledge base in relational database. It is proposed that a metadata model of ontology is the essential element for ontology design of knowledge. The extended entity relationship model is also offered as the design method for modeling structural knowledge. Finally, the knowledge of data integrity of relational database is adopted as the example to implement a simple ontology-based knowledge base on Oracle 11gXE database for verification. [7]

The objective of this thesis is to identify new ways to build ontologies from textual corpora automatically. A basic challenge we identify is that the ontology to a large extent is absent from set of domain specific texts because it forms part of the background knowledge the reader brings to the act of reading. The knowledge taken for granted by the writer is rarely made explicit, and thus it is computationally inaccessible to a significant degree.

Basically huge literature is present in text format. To extract knowledge from it and management of this huge database is also important. So here building an ontology from text document i.e. need to extract concepts and relations among these concepts.

III. PROBLEM STATEMENT

The Enormous amount of information is available on semantic web. Many e-learning and knowledge database are present and will up come in future where semantic webs are used. So here Ontology is one of most important field in semantic web applications. Ontology creation varies from domain to domain. Large amount of information is available in text document. Now extraction of information's from this to build ontology is one of challenge. In this paper we are presenting the architecture, which will build the Ontology from text document by extracting concepts and relationship among this concepts.So this huge text collection on semantic web given rise to the questions like:

1. How to manage this huge amount of text data so that we get right information on search?
2. How to convert this unstructured database to structured database?
3. How to perform search on this database to get accurate results?
4. How to convert process of Ontology creation from Manual to semi-automatic?

Ontology can be used for more specific search and information gain in systems like knowledge base, e- learning. We are proposing a semi-automatic architecture to extract the knowledge from text files like .doc, .txt, .pdf . And it will be represented an ontology in terms of classes, concepts and relation. Here we are mainly focusing on domain ontology. We are taking into consideration some particular domain like education where users need to share the information commonly. And the search engine which is developed is Apache Lucene search engine which is a Indexed based keyword search engine. The search engine will show the keyword related documents present in database.

IV. ARCHITECTURE OF SYSTEM

Here presenting architecture which will produce ontology from the text documents. There are certain obligatory steps which will lead into ontology building. Fig 3 shows the architecture model.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

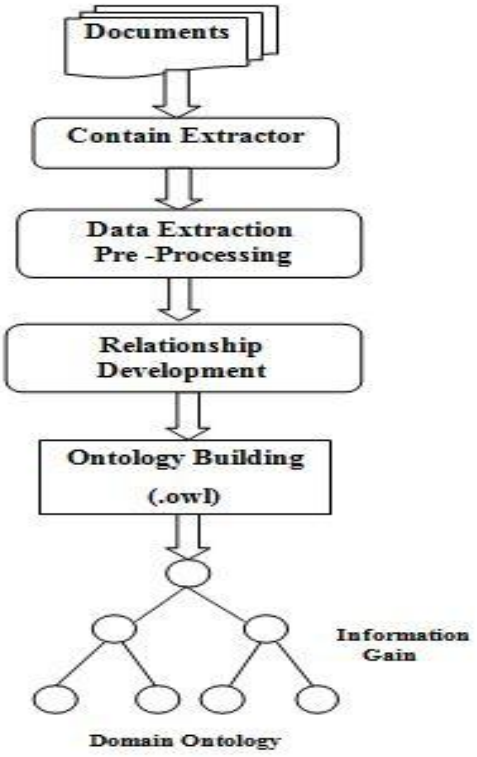

Fig.1 Architecture Diagram

Input: We are considering education domain for building Ontology. Large amount of literature, information is available in the text format like .doc,.txt,.pdf,.ppt. So for this system as input we are considering text document.

Document Extractor: As explain above information is present in heterogeneous document formats. Before processing further we need process every input document. Now document extractor check with the type of document uploaded and respective format libraries are called to extract contains.

Pre-processing: In Pre-Processing module certain algorithms will be running for Identification of Part of Speech (POS), TFIDFConceptExtraction, TFIDFInstanceExtraction, stemming operations, Stopword eliminations. Identification of POS like noun, pronoun, adjective, adverb etc. from text document will be done. In stopword list, there are some words like the, a, an, which will not bring up any information are removed. From the list of stopword list the identified stop words are removed from the input document.

Relationship development: In this step will be looking for classes and subclasses from the extracted contains i.e. concepts and instances. Putting the concepts and Instance under structure of ontology is an important task. Here the nouns are considered as concepts while pronouns are consider as Instances. Now Instances are assigned to subclass while nouns will be main classes of .owl file. Also in .owl the structure called Document Pointer the implemented as data properties which will point the document name from which document this concept or Instance is taken. Document Pointer also have Object property refer to which shows the domain and range of particular concept or Instance.

Ontology Building: Finally classes, concept and relation between these concepts are formed i.e. in this step ontology will build in.owl file format. These class, concepts are having association among them it may be one to one, one to many or many to many. The output from all above step will lead into ontology building from text document. This can be used for different knowledge base to represent and maintain the database, also to improve search, Information retrieval.

V. EXPERIMENTAL RESULTS

Here we have shown the architecture for ontology building from text document. In following figures actual implementation results can be seen. Figure 1 show the input window which will take the input as folder containing text files. The text files can be .pdf,.txt,.doc.

Figure 2 shows after uploading the document to the ontology builder pre-processing is done in which concepts are getting extracted using TFIDF concept Extractor.

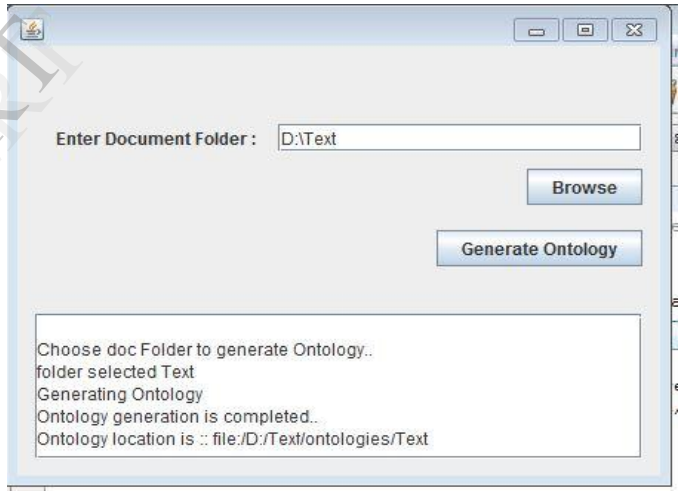

Fig.2 Input the Document

Figure 3 shows the actual generated .owl file from the input text document. The structure of owl file shows RDF Id s like Entity, Concept, Instances, Subclasses and Document Pointer etc. Unique URI for each RDF Id is created.

IJERT

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

```
ALGORITHM_CONTROLLER.execute: TFIDFConceptExtraction
ReferenceManager.getLocalStore: creating new DocumentReferenceStore for TFIDFConceptExtraction
AbstractEntityExtraction.getEvidenceChanges: file:/D:/Text/Elsevier.txt, type: 0
Concept :: power network
Concept :: network state
Concept :: power transfer
Concept :: function
Concept :: power transmission
Concept :: network
Concept :: voltage control
Concept :: power system performance
Concept :: system
Concept :: power flow
Concept :: device
Concept :: minimise
Concept :: implementation
Concept :: distribution
Concept :: system voltage profile
Concept :: oscillation
Concept :: wind generator
Concept :: paper
Concept :: compensator
Concept :: flicker
Concept :: voltage profile
Concept :: superiority
Concept :: result
Concept :: state
Concept :: phenomenon
Concept :: load
```

Fig.3 Extracted Concepts from Document

Figure 4 shows the actual search engine based on Lucene. Apache Lucene is an Indexed based search engine. Here using this key word based search engine is developed. Enter a keyword in text box and submit the query. As result of query fired will get the list of document names in which that word is present.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE rdf:RDF [
    <!ENTITY owl 'http://www.w3.org/2002/07/owl#'>
]>

<rdf:RDF
    xml:base="http://www.elearning.org/ontology"
    xmlns:a="http://www.elearning.org/ontology#"
    xmlns:owl="http://www.w3.org/2002/07/owl#"
    xmlns:owlx="http://www.w3.org/2003/05/owl-xml#"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">

<owl:Ontology rdf:about=""/>

<owl:Class rdf:ID="Concept">
    <rdfs:subClassOf rdf:resource="#Entity"/>
</owl:Class>

<owl:Class rdf:ID="DisjointClasses">
    <rdfs:subClassOf rdf:resource="#Element"/>
</owl:Class>

<owl:Class rdf:ID="DocumentPointer">
    <rdfs:subClassOf rdf:resource="&owl;Thing"/>
</owl:Class>

<owl:Class rdf:ID="Element">
    <rdfs:subClassOf rdf:resource="&owl;Thing"/>
</owl:Class>

<owl:Class rdf:ID="Entity">
    <rdfs:subClassOf rdf:resource="#Element"/>
</owl:Class>

<owl:Class rdf:ID="Instance">
    <rdfs:subClassOf rdf:resource="#Entity"/>
</owl:Class>

<owl:Class rdf:ID="InstanceOf">
    <rdfs:subClassOf rdf:resource="#Element"/>
</owl:Class>

<owl:Class rdf:ID="Relation">
    <rdfs:subClassOf rdf:resource="#Element"/>
</owl:Class>
```

Fig.4 Output .owl file

Figure 5 shows the experimental comparative result on owl file. There are two methods present to extract information from .owl or RDF file. The SPARQL is traditional language which is used in many existing system. Here we have implemented both the approaches SPARQL and Apache Lucene. The resultant graph shows the performance chart of both approaches. Like how much time is taken by SPARQL and Lucene to generate result from certain size of databases. So the graph analysis shows the time taken by Lucene search engine is less than SPARQL.

Figure6. Shows the experimental results for given query search. Here using straightforward evaluation measures derived from the information retrieval community, namely, precision and recall. Precision can be seen as a measure of exactness or fidelity, whereas recall is a measure of completeness.

$$\text{Recall} = \frac{\{\text{Relevant documents}\} \cap \{\text{Retrieved documents}\}}{\{\text{Relevant documents}\}}$$

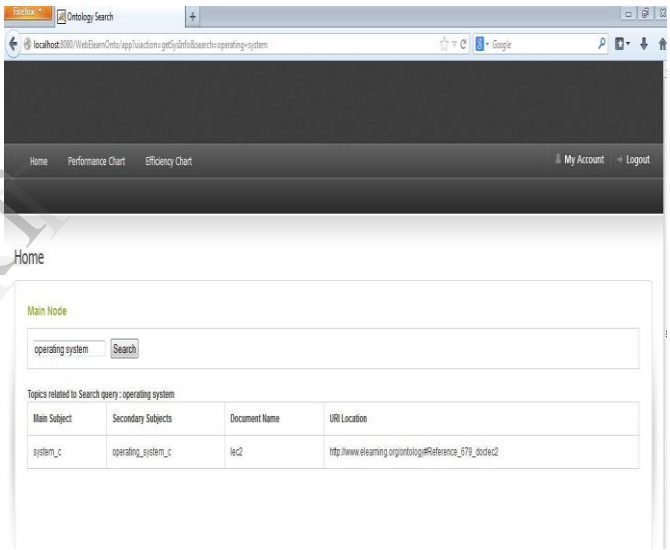

Fig.5 Lucene Search engine

IJERT

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

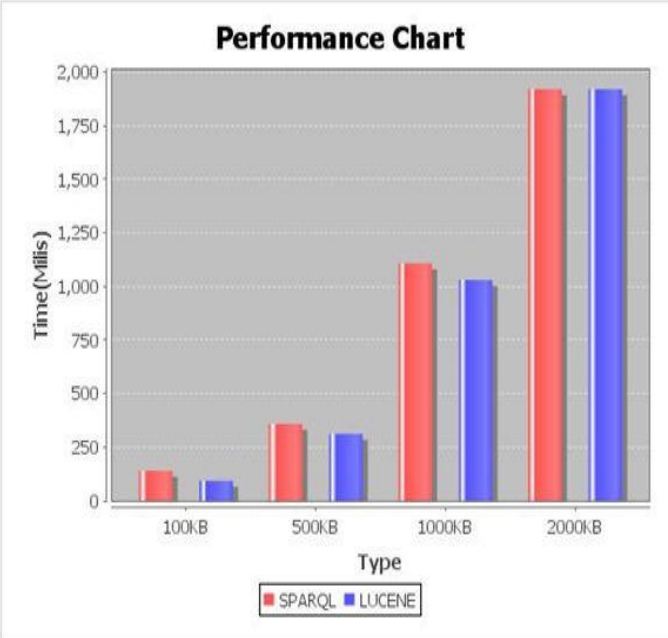

Fig.6 Performance chart

Fig.7 Efficiency chart

So here efficiency of system is shown by efficiency chart. According to recall formula values are generate and graph is plot for how many number of documents are retrieved by SPARQL and Lucene search engine for different input query. So graph shows the maximum no of document are search by Lucene engine.

VI. CONCLUSIONS

Here we have shown the architecture for ontology building from text document. This will reduces the manual efforts and time for building ontology. And also develop the search engine which will be solving the problem of keyword based search on ontology. Finding appropriate sense of terms is more challenging task. Making full automation, Ontology mapping are further challenges.

VII. ACKNOWLEDGMENTS

Wish to thank our colleagues for their precious advices on our work and for their contribution in reading over this article. Also will like thank to my parents and friends for their constant support and guidance.

REFERENCES

[1] Matteo Gaeta, Stefano Paolozzi, and Saverio Salerno "Ontology Extraction for Knowledge Reuse: The e-Learning Perspective" IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART A: SYSTEMS AND HUMANS, VOL. 41, NO. 4, JULY 2011.

[2] Maizatul Akmar Ismail, Ram Gopal Raj, S. Abdul Kareem "Dynamic Ontology Editor for a Knowledge Management System of Scholarly Activities" IEEE 2011 International Conference on Semantic Technology and Information Retrieval" 28-29 June 2011, Putrajaya, Malaysia 978-1-61284-353-7/11/$26.00 ©2011

[3] Leyla Zhuhadar, Olfa Nasraoui, Robert Wyatt, 2009 13th International Conference Information Visualisation 978-0-7695-3733-7/09 $25.00 © 2009 IEEE DOI 10.1109/IV.2009.47

[4] Petr K˘remen and Zden˘ek Kouba "Ontology-Driven Information System Design", IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C: APPLICATIONS AND REVIEWS, VOL. 42, NO. 3, MAY 2012

[5] Joel Villanueva Chávez, Xiaoou Li "Ontology based ETL process for creation of ontological data warehouse"

[6] Ayesha Ameen , Khaleel Rahman Khan, B.Padmaja Rani , "Construction of University Ontology" IEEE World Congress on Information and communication Technology ,2012

[7] Huei-Zhen Gu "Research on Building Computation of Ontology-Based Knowledge Base" 2012 12th International Conference on ITS Telecommunications 978-1-4673-3070-1/12/ ©2012 IEEE 437

[8] FE´ LIX GARCI´ A, FRANCISCO RUIZ, CORAL CALERO,"Effective use of ontologies in software measurement" The Knowledge Engineering Review, Vol. 24:1, 23–40.

[9] Trupti Hake and V.S.Deshmukh; "Building of domain ontology using NLP" in International Conference on Recent Trends in Communication and Computer Networks - ComNet 2013. RE588

[10] M.H. Abel, A. Benayache, D. Lenne, C. Moulin, C. Barry, and B. Chaput. Ontology-based Organizational Memory for e-learning. Educational Technology & Society, 7(4):98–111, 2004.

[11] Christopher Arthur Brewster "MIND THE GAP: BRIDGING FROM TEXT TO ONTOLOGICAL KNOWLEDGE" Ph.D thesis , University of Sheffield July 2008.

IJERT

(This work is licensed under a Creative Commons Attribution 4.0 International License.)