Exploration on Service Matching Methodology Based On Description Logic using Similarity Performance Parameters

K.Jayasri
Final Year Student
Department of CSE
IFET College of engineering
Villupuram-TN, India
nishajayasri@gmail.com

R.Rajmohan
Assistant Professor
Department of CSE
IFET College of Engineering
Villupuram-TN, India
rjmohan89@gmail.com

D.Dinagaran,
Assistant Professor
Department of CSE
IFET College of engineering
Villupuram-TN, India
ddinagaran@gmail.com

Abstract— Extracting accurate information for the user input is the main drawback for information retrieval system in web. To overcome the matching problem, a description logic based matching technique is proposed. The description logic consists of TBox and ABox to separate the individual and universally quantified properties. The user input can be matched against ontology database and allows us to formalize information of domain as classes and instances. The DL-structured data format is processed through Hadoop to obtain the optimal matching service platform. The unstructured information is stored as files in the Hadoop Distributed File System and MapReduce operation is performed for simplified development of programs. The SPARQL query is used for querying data stored in Mongodb. Mongodb has collection of RDF datasets and are processed for matching using SPARQL query. The resultant dataset for the user input is given in the form of text document. The performance of the matching system is improved in terms of precision and recall value. Comparing with the existing system proves that the proposed system provide an optimal solution with minimal search time and maximum accuracy.

Keywords—ontology matching; description logic; hadoop; mongodb; SPARQL query language.

I. Introduction

The Web Ontology Working Group of W3C found that RDF cannot fully satisfy all the requirements of the semantic web. To overcome this problem, efficient processing of ontology model is used. The ontology is used to formalize data based on the domain and describe the relations. The OWL format is used to find the domain of knowledge and it is the best method to describe the semantic web. The user input is processed through ontology database and it creates the structure based on the resources in the OWL model. The obtained semantic knowledge is matched with the set of databases in the ontology database. The database has collection of information about the domain and the matched information for the user input is obtained in structure format. The OWL-DL format is obtained using description logic to obtain the RDF framework model. To reduce the datasets stored in HDFS (Hadoop Distributed File System), the MapReduce framework is used.

II. RDF AND SPARQL

Efficient model for storing and representing data by Resource Description Framework (RDF) plays an important role to retrieve the data. To convert the structured data to machine readable form, the RDF framework [1] is used. RDF is a directed, labeled graph and data model used for semantic web data management. Each statement of RDF has subject, predicate and object are denoted as triple. The resources are described by using statement of RDF and all resources have a

URI for their unique representation. An RDF Graph Representation for the resources is used to create the SPARQL query.

The SPARQL [2] is the query language to retrieve the data from an RDF. The SPARQL query language is created and matched with the URI and generate the new RDF graph. For the new RDF graph, the SPARQL query is created and MapReduce jobs are performed.

IJRD International Journal of Research & Development Organisation
ISSN - 3785 - 0855
JOURNAL OF COMPUTER SCIENCE AND ENGINEERING

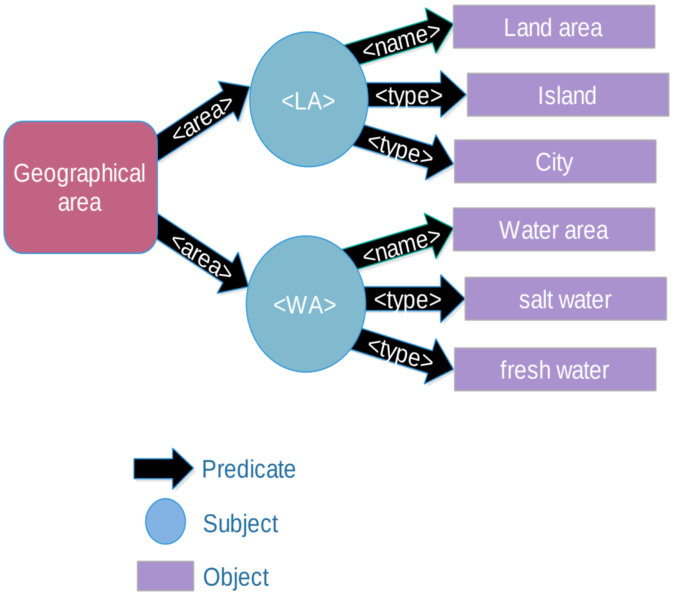

Figure 1.Resource Description Framework (RDF) graph.

III. Description Logic

Description Logic [7] is a family of knowledge representation language. The knowledge of an application domain is represented by first defining the relevant concepts of the domain and then using the concepts to specify properties of objects and individuals occurring in a domain. DL is constituted by two components such as TBox and ABox. The TBox describes a set of universally quantified assertions using general properties of concepts and roles. The ABox comprises assertions on individual objects.

In our approach, the structured data is processed through description logic to obtain the OWL-DL [9] language format. The OWL-DL language has based on the RDF to describe the semantic meaning. The proposed logic includes syntax and semantics for conversion of OWL Axiom to DL Axiom and an RDF framework [8] to convert the structured data to machine readable form. The RDF model mainly used to describe the conversion of individuals as rdf:description, rdf:type and rdf:property.

IV. Hadoop

In order to effectively handle and store the large amount of RDF data, the Hadoop framework is used. Hadoop [11] is an open source implementation that enables the distributed processing of large datasets across cluster of commodity server. It is capable to connect and coordinate thousands of nodes inside a cluster. The Hadoop framework has two components such as HDFS [5] and MapReduce [12]. HDFS is designed for data storage and MapReduce for data processing.

Hadoop Distributed File System (HDFS) store the RDF data [6] based on the two nodes such as Data Node and Name Node. In our approach, the DL data format is based on the RDF data and we cannot directly store the RDF data in HDFS. To overcome this problem, the N-triple format is taken [3] and split into predicate files based on the type of object. It is stored as text files according to the schema in HDFS. An important feature of HDFS is data replication and each file is divided into blocks and replicated among nodes. The files have the collection of data and it is stored in Data Node. The Name Node describes the location of the file using the URI and manages the file namespace. The SPARQL query [15] is created for the data stored in HDFS and MapReduce operation is performed.

V. MapReduce

Map Reduce is used to execute the SPARQL query and provide the parallel processing over a large number of nodes to simplify the data [10]. The SPARQL query created using the DataNode and matched with the NameNode and generates the RDF graph. Finally, the SPARQL query language is created for the RDF graph and the data is retrieved from the HDFS.

The Mapstep has two nodes such as master node and worker node [4]. The master node split the data and assigns the keyvalue pairs. The master node picks the idle workers and assigns each one a map tasks to set the intermediate keyvalue pairs and send back to master node. The master node stores the details about the location of the data. The Reduce function takes the intermediate keyvalue pairs and reduces to a smaller solution. Based on the smaller solution, again the SPARQL query language is created.

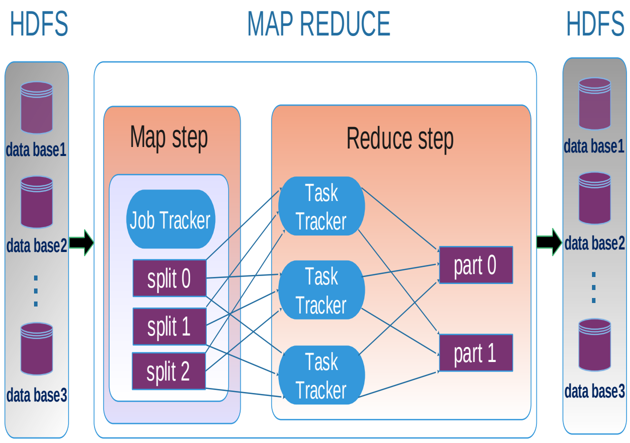

Figure 2. HDFS and MapReduce flow

VI. Proposed Architecture

The proposed architecture explains about the systematic study on the performance of service matching methodology

IJRD International Journal of Research & Development Organisation
ISSN - 3785 - 0855
JOURNAL OF COMPUTER SCIENCE AND ENGINEERING

based on description logic. Our architecture consists of several components for storing and querying an RDF data. The description logic converts structured data into DL format based on RDF framework. The resultant is processed through the Hadoop component to generate the SPARQL query language. The query language is matched against Mongodb and gives the output.

In the architecture, the unstructured user input is converted to structure format by matching the user input to the ontology Database and formalize information as classes. The user input is converted to structure format by matching the user input to the ontology database and formalize information as classes and instances. The description language converter uses the TBox and ABox to describe the concept definitions, inclusions and assertions. The OWL-DL axiom based on RDF framework to formalize data in machine understandable form. To store and query RDF data, the Hadoop framework is used. This component has two subcomponents such as Hadoop Distributed File System (HDFS) and MapReduce. The DL format is processed through Hadoop component and stored as text files in HDFS and MapReduce operation is performed. The reduced dataset is stored in HDFS and SPARQL query language is created. The Mongodb has the collection of RDF data and it matched with the SPARQL query and gives the output in the form of text document.

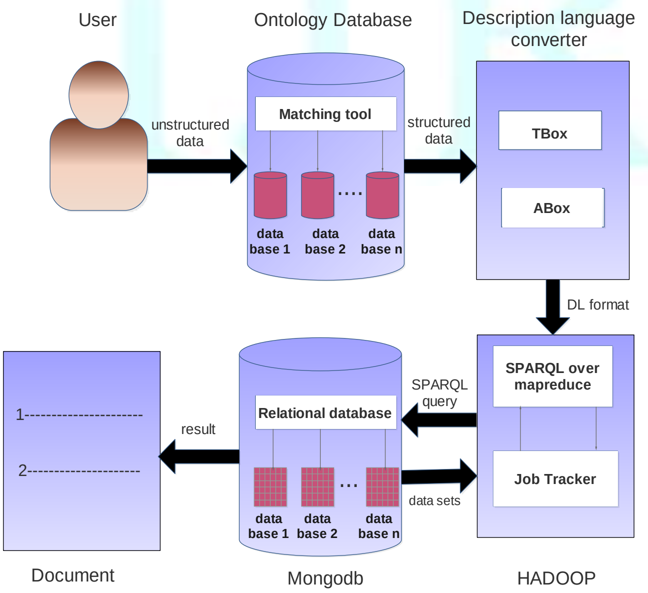

Figure 3: Architecture of Sematic similarity based matching

VII. Experimental Setup

Our experiments use the Windows 7 operating system with Pentium 4 processor, 1-Gbyte RAM with a clock speed as 2.7 GHz. The capacity of the Hard disk drive is 200GB. The tools and database such as Hadoop 0.18.10 and Mongodb are installed in the system. Our approaches are implemented in Core Java language with the version JDK 1.7 and running in Netbeans IDE 7.2.1. The RDF datasets are stored in Mongodb and retrieved using the SPARQL query language.

VIII. Datasets and Queries

Table 1: Query Sets

ID	Query	Shape
Q1	SELECT ?x WHERE { ?x geog:hasCapital ?y }	Point
Q2	SELECT ?s ?p WHERE { ?s geog:isLowestPointOf. geog:Assam. }	Point
Q3	SELECT ?m ?pWHERE { ?p rdf:type ex:Park. ?m rdf:type ex:Monument. geog:within ?p }	Graph

Q4	SELECT ?p WHERE {
	?p rdf:type ex:Park
	geog:hasGeometry ?pgeog.
	ex:NewyorkMonument
	geog:hasGeometry ?mgeog.
	FILTER (geogf:distance (
	?pgeog, ?mgeog, units:m)<3000
	}
		Graph

Q5	SELECT ?m ?p WHERE { ?m rdf:type ex:Monument geog:hasGeometry ?mgeog. ?p rdf:type ex:Park geog:hasGeometry ?pgeog. ?m geog:within ?pgeog. }	Graph

The experiments were based on the geographical datasets. We generated the geographical datasets and expressed in OWL format with the ontology type in RDF model. We collected the 9 RDF class and 14 properties. The RDF triples are collected for the datasets to express the OWL-DL format.

The queries from the user are expressed in OWL to find the semantic relation. For the purpose of our experiments, we use the SPARQL query language. It's because our experiments tested performance for extracting accurate information for the queries using ontology matching. Sample queries used for our evaluation are shown in the above Table1: Query Sets.

IX. Experimentation

Our experiments based on the semantic search for Information Retrieval (IR). In search engine, the IR is based on the keyword and return less relevant documents. The proposed search methodology uses the ontology matching technique and MapReduce to develop the efficient query plan for Information Retrieval. The ontology model gives the relevant RDF datasets based on semantic similarity. By applying MapReduce we can join the relevant RDF datasets in a specific reducer. Therefore, the relevant information is retrieved accurately in search engine for the user input.

IJRDO International Journal of Research & Development Organisation
ISSN - 3785 - 0855
JOURNAL OF COMPUTER SCIENCE AND ENGINEERING

X. Precision and Recall

The Precision and Recall [13] are commonly used to measure the accuracy of an information retrieval system. Precision is the fraction of retrieved webpage that are relevant to the search. Recall is the fraction of the webpage that are relevant to the query that is successfully retrieved. They are computed as,

$$\text{Precision} = \frac{|\text{relevant webpages} \cap \text{retrieved webpages}|}{|\text{retrieved webpages}|}$$

$$\text{Recall} = \frac{|\text{relevant webpages} \cap \text{retrieved webpages}|}{|\text{relevant webpages}|}$$

The precision and recall are combined to calculate the F-measure. The F-measure [14] is the harmonic mean and used in IR for measuring search efficiency.

$$\text{F} - \text{measure} = 2.\frac{\text{precision.recall}}{\text{precision} + \text{recall}}$$

Here, ranking is analyzed by Information Retrieval (IR) accurately using ontology matching models and MapReduce for the user input.

XI. Evaluation

The performance of the OWL-DL model and MapReduce are used to develop the query plan for the user input. Based on the semantic relation in query plan are used to extract the accurate webpage in search engine. The queries based on geographical datasets and the precision, recall and F-measure is computed.

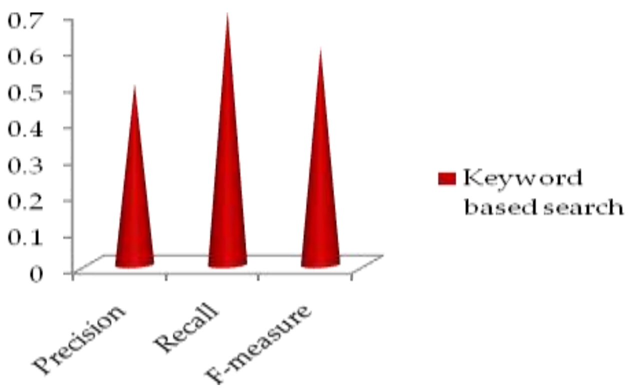

Figure 4. Experimental results for keyword based search

In Figure 4, the Information Retrieval (IR) based on the keyword and gives less relevant web pages for the queries from the user.

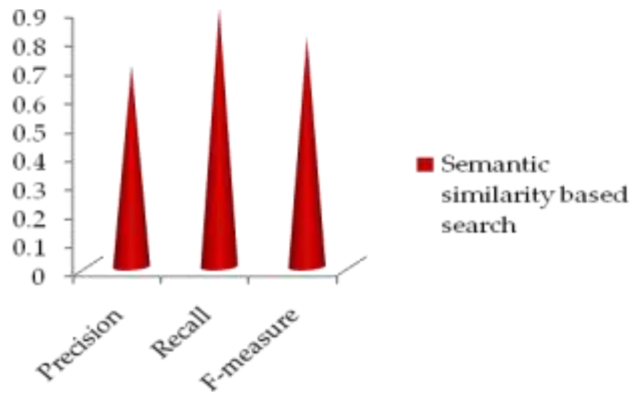

Figure 5. Experimental results for semantic similarity

From the Figure 5, the semantic similarity based search using ontology matching model gives the efficient query plan and retrieve the accurate information using the SPARQL query.

XII. Conclusion and Future Work

The proposed information retrieval system overcomes the existing matching problem with the help of description logic. The TBox and ABox components are used to formalize the semantic information from the user input with the help of ontology. MapReduce operation is performed in Hadoop for simplifying the querying and matching process. The SPARQL query is used for querying the DL structured data stored in Mongodb. The performance of the proposed system exceeds the existing in terms of performance measures such as precision, recall and F-measure.

Our future work includes extending our DL pattern based matching by integrating Hyperclique and Lattice pattern.

References

[1] Wangchao Le, Feifei Li, Anastasios Kementsietsidis, and Songyun Duan, "Scalable Keyword Search on Large RDF Data," IEEE, Knowledge and Data Eng., vol. 26, no. 11, 2014.

[2] LIU Chang, WANG Haofen, YU Yong, XU Linhao, "Towards Efficient SPARQL Query Processing on RDF Data," vol. 15, no. 6, 2010, pp. 613-622.

[3] Craig Franke, Samuel Morin, ArtemChebotko, John Abraham, and Pearl Brazier, ".Efficient Processing of Semantic Web Queries in HBase and MySQL Cluster," 2013.

[4] Dawei Jiang, Anthony K. H. Tung, and Gang Chen, "MAP-JOIN-REDUCE: Toward Scalable andEfficient Data Analysis on Large Clusters," IEEE, Knowledge and Data Eng., vol. 23, no. 9, 2011.

[5] Praveen Kumar, Dr Vijay Singh Rathore, "Efficient Capabilities of Processing of Big Data using Hadoop Map Reduce," vol. 3, issue 6, 2014.

[6] M. Husain, J. McGlothlin, M. M. Masud, L. Khan, "Heuristics-Based Query Processing for Large RDF Graphs Using Cloud Computing," Knowledge and Data Eng., vol. 23, issue 9, 2011.

[7] Krotzsch .M, Simancik .F, and Horrocks .I, "Description Logics," Intelligent System, vol. 29, issue 1, 2014.

[8] Jeff Z. Pan and Ian Horrocks, "RDFS (FA): Connecting RDF(S) and OWL DL," Knowledge and Data Eng., vol. 19, no. 2, 2007.

[9] Pan. J. Z, "A Flexible Ontology Reasoning Architecture for the Semantic Web," Knowledge and Data Eng., vol. 19, no. 2, 2007.

[10] Bo Liu, Keman Huang, Jianqiang Li, MengChu Zhou, "An Incremental and Distributed Inference Method for Large-Scale Ontologies Based on MapReduce Paradigm," Cybernetics, vol. 45, issue 1, 2015.

[11] Jun Liu, Feng Liu, N.Ansari, "Monitoring and analyzing big traffic data of a large-scale cellular network with Hadoop," Network, vol. 28, issue 4, 2014.

[12] Cheng Chen, Zhong Liu, Wei-Hua Lin, Shuangshuang Li, Kai Wang, "Distributed Modeling in a MapReduce Framework for Data-Driven Traffic Flow Forecasting," Intelligent Transportation Systems, vol. 14, issue 1, 2013.

[13] Viviana Mascardi, Angela Locoro, and Paolo Rosso, "Automatic Ontology Matching via Upper Ontologies: A Systematic Evaluation," Knowledge and Data Eng., vol. 22, no. 5, 2010.

[14] Pavel Shvaiko and Jerome Euzenat, "Ontology Matching: State of the Art and Future Challenges," Knowledge and Data Eng., vol. 25, no. 1, 2013.

[15] R. Valencia-Garcia, F. Garcia-Sánchez, D. Castellanos-Nieves, J. T. Fernández-Breis, "OWLPath: An OWL Ontology-Guided Query Editor," Systems,Man and Cybernetics, vol. 41, issue 1, 2011.