ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

DYNAMIC QUERY FORMS FOR ANALYZING THE RDF AND OWL FILES - IMPLEMENTATION DETAILS

Ms. Aruna M.Gunje*, Dr.Suhas Raut

* Master of Engineering student, Department of Computer science & Engineering,
NK Orchid College of Engineering & Technology,
Maharashtra, India

Ph.D. Professor, Department of Computer science & Engineering Science and Engineering,
NK Orchid College of Engineering & Technology,
Maharashtra, India

ABSTRACT

The main issue of the Query Form is that the interactions among the users and query result. The rapid development of web information and scientific databases, modern databases become very vast and complicated. Therefore, it is very difficult to design a static query forms to assure various ad-hoc database queries lying on those complicated databases. The Dynamic Query Form is one of the mostly used curious database query form interfaces for querying database, which is dynamically create query forms. This paper proposes a novel approach that curious database information seeking that may enable users to cope with information overload. It generates an iterative process and performed repeatedly until user is satisfied with the query result analysis. This proposed system mainly focuses on dynamic generation of query form which analyzes the information within the web ontology files with minimum time duration. It is able to dynamically produce query forms, for both relational and non-relational (unstructured) data.

KEYWORDS: Query form, query form generation, user interaction, RDF, OWL.

INTRODUCTION

The predefined old query forms are not able to satisfy various queries that are executed by a user on those databases and are not enough to solve the query problems. Also it is difficult to design set of static query forms to satisfy various database queries on these complex databases if a user is unknown with the schema format of particular database in advance, what he or she wishes from it? It will confuse him/her. Writing good structured queries can be challenging due to a number of reasons, including the user's lack of informality with the query language and the user's ignorance of the knowledge of schema. In conventional system people access the information using SQL but in modern databases it is very difficult and complex. So information retrieval is very difficult and does not access fast. Suppose there are two tables in database employee and company details both tables are correlated to each other and manager wants to access the information about their employee's past education at that time manager fires the query in SQL syntax and to understand this syntax is very difficult for the non-technical user who unknown with query language. The dynamic query form query interface is a new approach for designing the dynamic query form for to analyze the data within web ontology files, which to select the particular file documents from database since it provides to data users with knowledge of instead of that document or the database schema. Traditional query forms are designed and pre-built by developers in information management Systems. It is difficult to design a set of static query forms which works on only the relational database to satisfy various ad-hoc database queries on complex databases. However, the creation of dynamic queries totally depends on user's manual editing. The approach of this paper is to propose a Dynamic Query Form, is an interactive tool which is capable of dynamically generating query forms for users. This DQF is to capture user interests during user interactions and to execute the query form iterative. The basic query form is upgrade iterative via the interactions between the user and proposed system until the user is satisfied with the query results. The system also provide user to add new document which is he/she want to search and analyze that one. This dynamic query form are works on unstructured data and user gets output with minimum time duration.

RELATED WORK

The users are those working on database are having easy work because of the query forms. In "Dynamic Query Forms for Database Queries" in this paper Liang Tang, Tao Li proposes that if before the query forms are designed by developers or DBA using predefined SQL statements. But as after development had done the databases becomes so complex that the DBA can't guess all the queries while developing query form [1]. S. Chaudhuri, G. Das, V. Hristidis, and G. Weikum, proposed that the query results are displayed by the query form. In this one the user interest is estimated based on the user's click through on query form. For example, if some data instances are clicked by the user interests. Then, the query form components which can capture these data instances should be ranked higher than other components. It introduces some notations and then defines expected precision and recall [2]. Wangchao Le and Feifei Li are do the experiments on both benchmark and large real RDF datasets show that our techniques are much more scalable and efficient in correctly answering keyword search queries for realistic RDF workloads than the existing methods [3]. Q.T.Tran, C.-Y.Chan and S. Parthasarathy proposed that the existing database clients and tools make great efforts to help developers design and generate the query forms, such as Easy Query [1], Cold Fusion [1], SAP, Microsoft Access and so on they provide visual interfaces for developers to create or customize query forms [5]. The problem of those tools is that, they are provided for the professional developers who are familiar with their databases, not for end-users. In given paper "Facetedpedia: Dynamic generation of query-dependent faceted interfaces for Wikipedia" C. Li, N. Yan, S. B. Roy proposes that the dynamic generation of the query dependent interfaces but user can not choice the option which he/she wants to enter in that one. [6] K. Chen, H. Chen, and N. Conway are proposed that the Dynamic Data Entry Form: Develops an adaptive forms system for data entry, which can be dynamically changed according to the previous data input by the user. Its work is different than dealing with database query forms instead of data-entry forms [7]. In this paper users generate automatically query form in advance. Here user enters the several keywords to find relevant query form. This leads to the conclusion that a query rewrite by mapping data values to schema values during keyword search. It displaying the returned form as a flat list [8]. In this paper proposes that, static checking technique for verifying the correctness of dynamically generated SQL query strings in database applications. This technique is based on applications of a string analysis for Java programs and a variant of the context- free language reach ability algorithm [9]. In Building dynamic faceted search systems over database, this paper is a domain independent system that provides effective minimum- effort based dynamic faceted search solution over enterprise database[10]. M.Jayapandian and H.V. Jagadish proposes that automatic approaches to generate the database query without user participation. It is a work-load driven model. It applies clustering algorithm to find representative queries. One of the disadvantage is that if user generate lots of query forms in advance, there are still more queries that cannot be satisfied by one of the query form [11]. In Building dynamic faceted search systems over database, it is a domain independent system that having effective minimum-effort based dynamic faceted search solution over enterprise database [12]. Query recommendation for interactive database exploration this paper is based on a collaborative approach to recommend database query components for database exploration, i.e. SQL queries as item in the collaborative filtering approach they recommend similar queries to related users. In this one problem arises that they do not have to good query result and recommendation is a query component for each iteration [13].This paper proposes that keywords are searched using the graphs which contain RDF dataset but it is long process to find out the keywords [14].

THE ARCHITECTURE

Following Figure 1.1 shows the proposed architecture. From above introduction and related survey in which there are some problem related to the dynamic query form. This proposed system are being developed to address various problems faced in the static query forms having with relational data therefore designing the dynamic query forms which gives most relevant search result and thereby overcoming the drawbacks of existing systems. Therefore to overcome various problems proposed system was developed in which instead of providing analyzed search data, proposed application displays results with efficient manner. Thus, dynamic query form application is designed to learn and analyze the data within the web ontology files. The concepts are modeled as ontology's, in order to capture the results within the files. This approach is most useful for to analyzing search queries submitted by user. The proposed system design a dynamic form which is useful for the person which having lack of informality with the query language hence he/she analyze the data in owl file and well understood the file.
Proposed system consists of following major activities:

1. Download the web ontology files (.owl) and copy the URL path:

User first downloads the .owl files from web and copies the URL path of that file and load the file. After loading the file ontology parser fetch ontology file with triples and parse that file and send to the controller. After getting

ontology file indexing that one and store index into the dataset.

2. Indexing web ontology files:

Controller indexing the .owl file i.e. it create index and send to database. The database contains files with having index number and sends the indexed data to the controller. The required data is sends to query module.

3. Query Module:

This is the backend process the queries are executed and get the results in this one RDF and OWL files are analyzed with user requirement. User fills up the query form and after that system executes the SPARQL query and displays the query results.

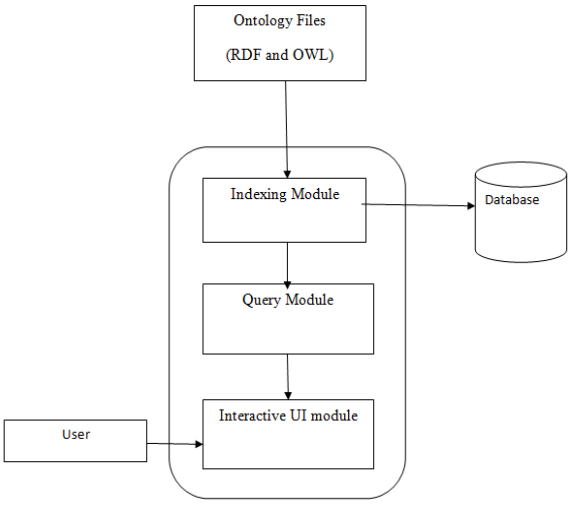


Figure 1 Overall System Architecture

4. Interactive UI Module:

In this module user sends the parameters to the UI module i.e. to fill up the query form .He/she selects particular file which wants to analyze and get the results with particular requirements. If user desire to analyze the new file which is not present in database, then user add new document into database. Copy the URL of that file and paste into textbox and load that file which is stored into database and get the particular results.

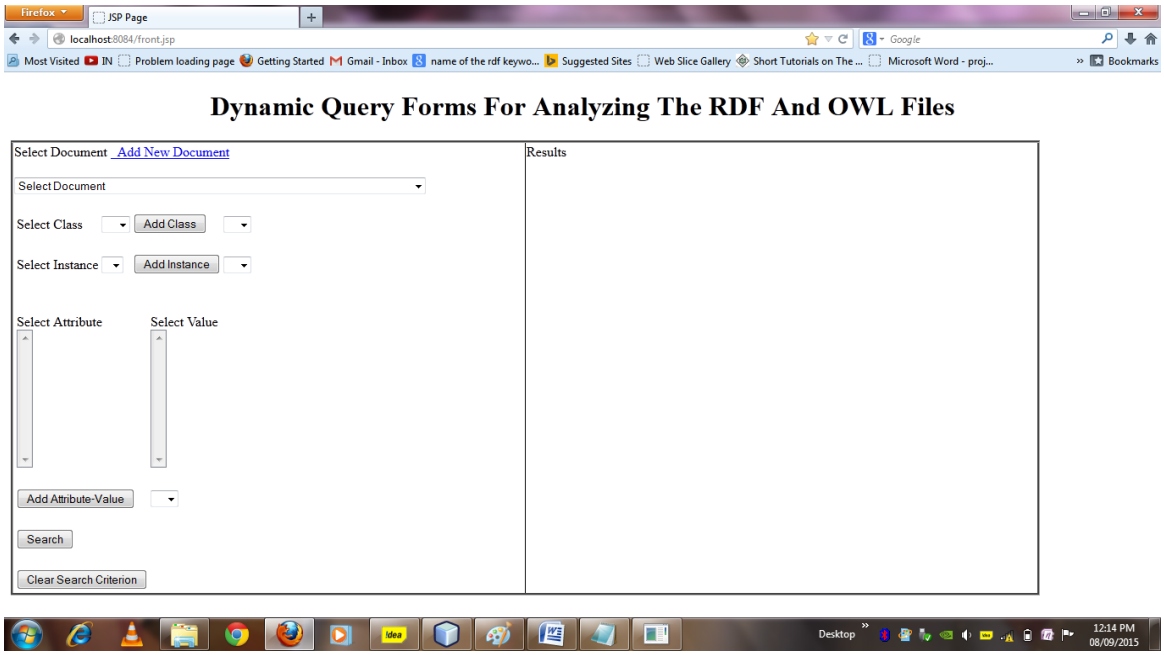
IMPLEMENTATION AND EXPERIMENTAL SETUP

In the experimental phase of the system first check that the URL which is given by the user are correct or not. The files are in proper format it contains the classes, instances and attributes with values. The first step is to select the particular documents which shown in given below GUI form. And click and select what the user wants. And add that particular values using the click through data. User gets the results having minimum time duration. This is one toolkit which analyzes the data within RDF and OWL files without writing the query.The dynamic query form is new approach that user fill up the form with his/her expectation and get the output quickly. If user is informal with query language then he/she does not obtain anything what will be wants? Then many questions are arises there. This is best interactive tool for that user who are familiar with query language but not perfectly. They obtain the results first and check the file data.Using this form they well understood the OWL file.

GUI Designs

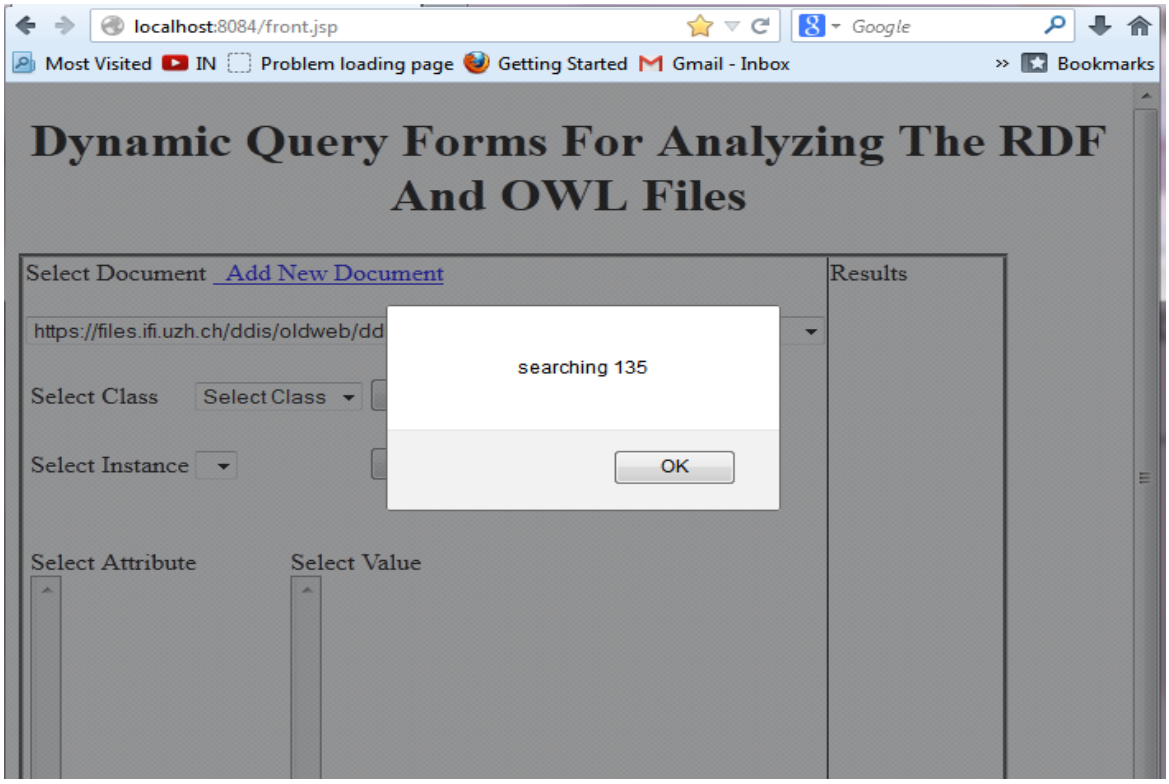
This is the basic dynamic query form front page which is shown in following screenshot 1. In this one document are the lists of URL path of the OWL files which are stored into the database. There one link that “Add New Document” if user want to add new file which is not in that list then he/she load that one into the database, and analyze the data and get the particular results. After that one select the class, add that class name ,then also select instaces and add instance name, add attribute value name. After selecting all component click on search button. If user wants to search new documents results then click on “clear search criteria” all are cleared which is selected before one. Then select new document which wants to select. This is the query form which is works on both relational and non-relational data. And it will overcomes the static form approach.

ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785

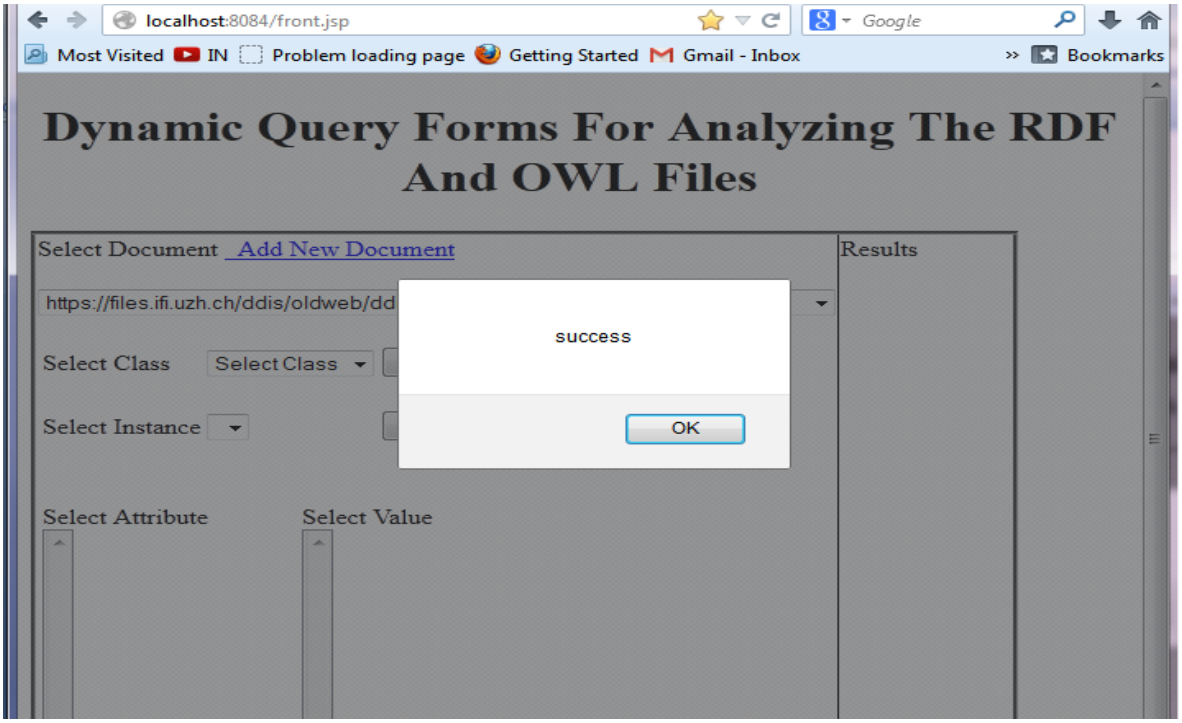


Screenshot 1 Dynamic Query Form

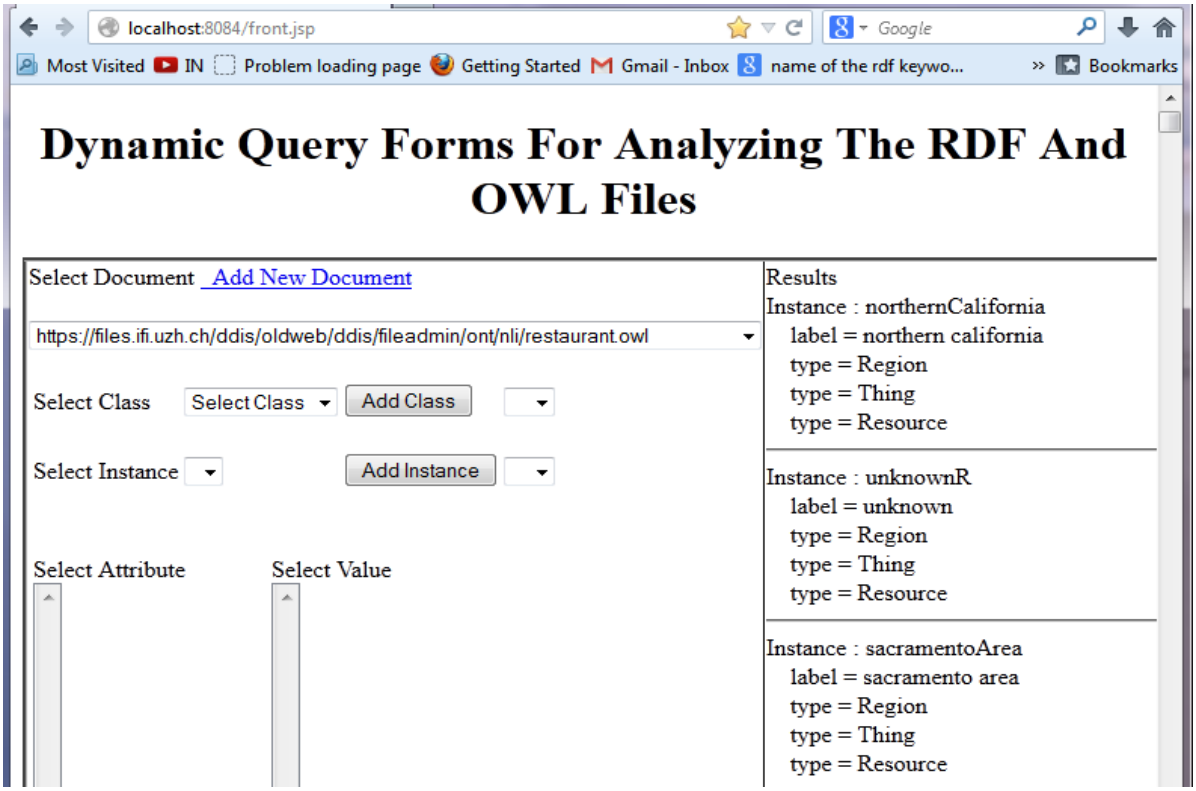
After selecting the particular documents system search that file with index number from database shown in following Screenshot 2. And successfully search that file in database shown in following Screenshot 3 and get the particular results that it contains all classes within that file as shown in following Screenshot 4.



Screenshot 2 File Index Searching



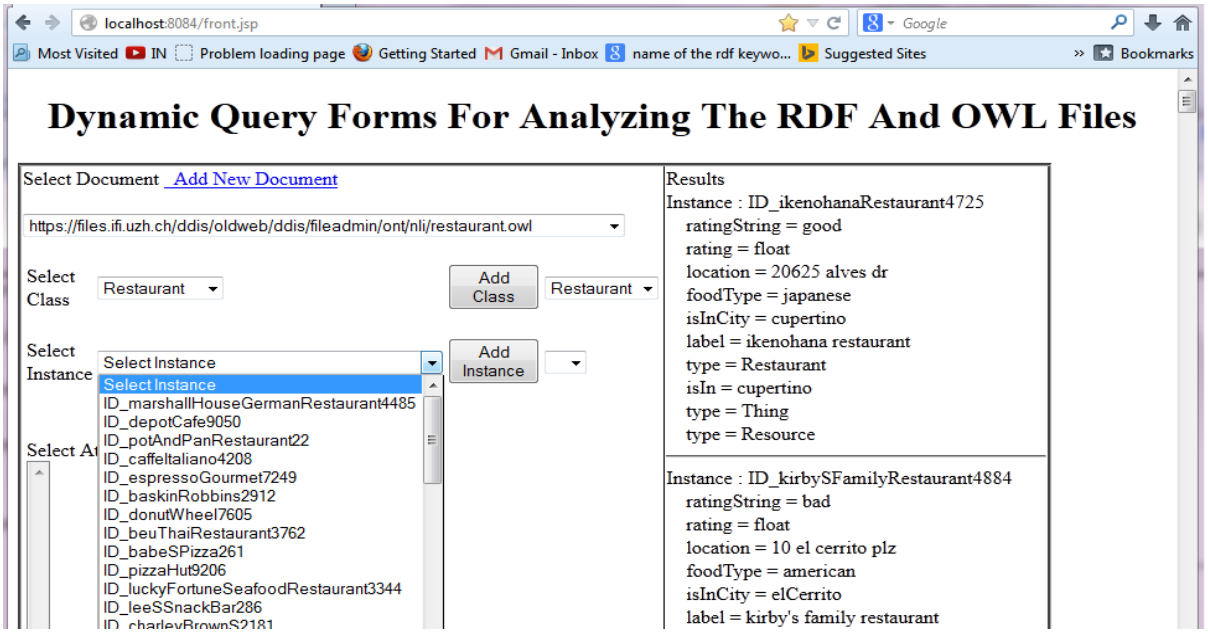
Screenshot 3 Searching File Successfully



Screenshot 4 Search Results

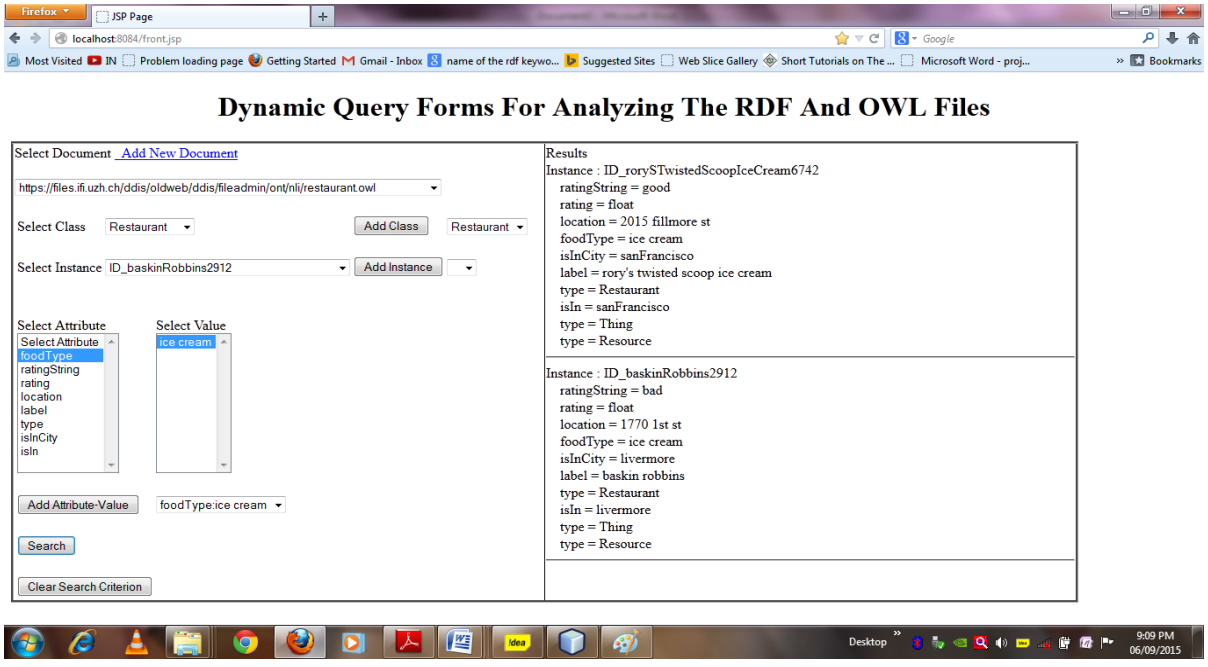
ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785

If user want to search the instances and data of only one class. Then select that class only then add class and get the all instances of that class which is as shown in following screenshot 5.This form displays the instances which is containing the class restaurant.



Screenshot 5 Searching One Class Instances

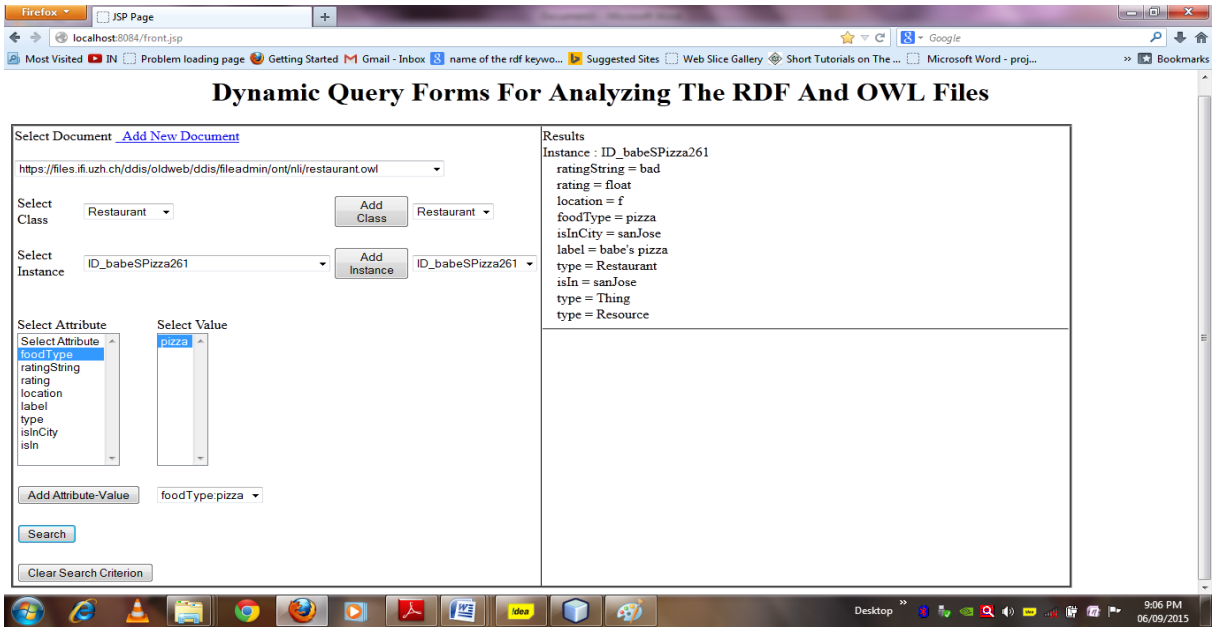
Following screenshot 6 which contain that, class having the many instances but display only that instances which having the same attribute value which is selected by the user. The following screenshot is having class is 'restaurant' and it contain many instances but search only that instances having the attribute value is "foodType:ice cream". It gets the 2 instances.



ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785

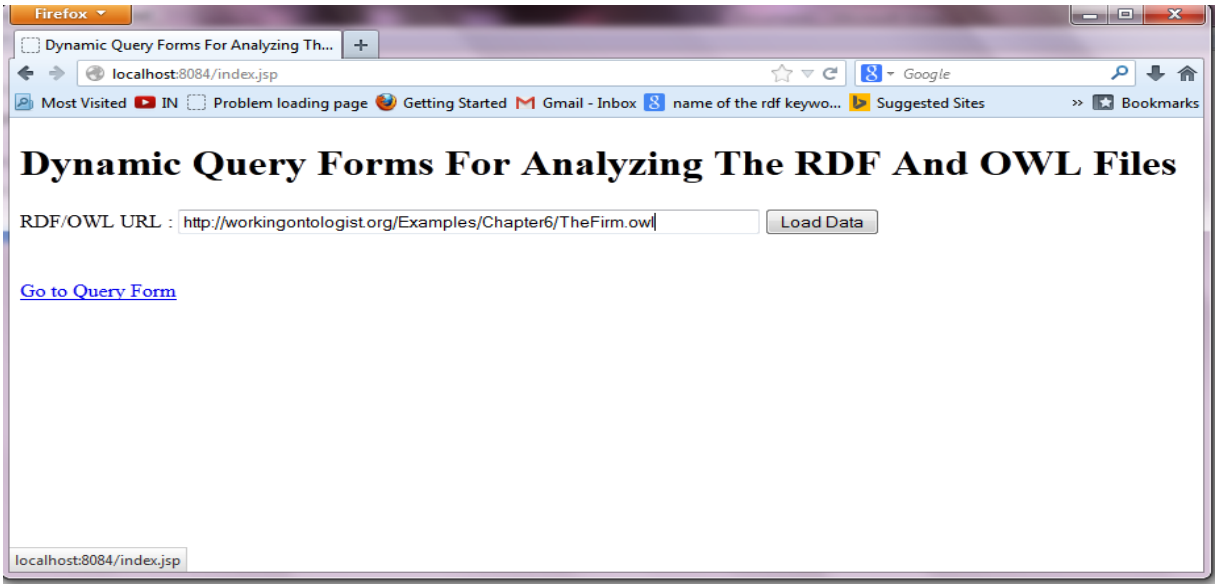
Screenshot 6 Searching Instances Having Same Attribute Value

Following screenshot 7 which displays only that instance which user desire. If user wants to search 'restaurant' class having instance 'ID-baseballRobber2912' and it has attribute value is "foodType: ice cream". It gets the selected instance result.



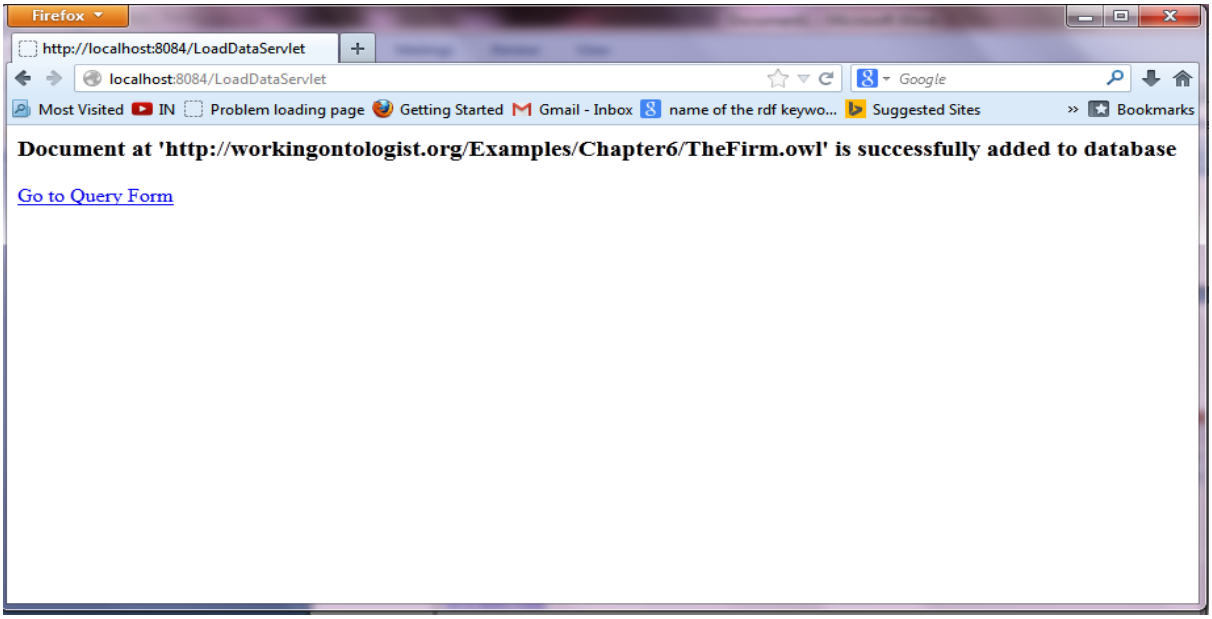
Screenshot 7 Searching Selected Instance Result

If user desire to add new document into database then click on "Add New Document" the control goes to load database page as shown in following screenshot 8. After that user paste the URL of OWL file and the load data into database.



Screenshot 8 Add New File In Database

Following screenshot 9 shown that user want to add "TheFirm.owl" file into database then just paste the URL path of that file and click on load data button. After that he get the message that file successfully added into the database.This is form which displays the result of what actual database changes and what containing in that one. If the file does not load properly then null document is added there. The system displays the message "null documents is added to database". If user check the database then he/she get the changes into the database with new value.



Screenshot 9 Database Result

Thus the proposed system achives the following advantages :
1] This dynamic Query Form is works on both relational and non-relational data.
2] It often leads to higher success rate and simpler query forms compared with a static approach.
3] Generate the query forms according to the user's desire at run time.
4] It is an interactive tool for analyzing RDF and OWL files efficiently.

CONCLUSION

The proposed dynamic query forms[DQF] is an innovative approach for searching OWL files and corresponding search results. It analyzes both the RDF and OWL file with the user interest having the minimum time duration and get the corresponding result for a user .This results are describes how the OWL file is made up and what it contain ? This proposed system is mostly used for those people who unknown with xml file format. Non-technical person can analyze the data in .owl file without query using click through data.

ACKNOWLEDGEMENTS

It was highly eventful at the department of Computer Science and engineering ,Nagesh Karjagi Orchid College Of Engineering Technology, Solapur. Working with highly devoted professor community will retain a memorable experience. Hence this acknowledgement is humble attempt to honestly thank all those who are directly or indirectly involved in my dissertation work and are of immense help to me. I would sincerely like to thank you my guide **Dr.Suhas Raut** for giving us perspective and taking interest in this dissertation work and whose advice and teaching helped me adopting a more pragmatic approach .

REFERENCES

[1] Liang Tang, Tao Li, Yexi Jiang, and Zhiyuan Chen,"Dynamic Query Forms for Database Queries" IEEE transactions on knowledge and data engineering,vol 26, no 9, pp 2166-2178, 2014
[2] S. Chaudhuri, G. Das, V. Hristidis, and G. Weikum, "Probabilistic information retrieval approach for ranking of database query results," ACM Trans. Database Syst., vol. 31, no. 3, pp. 1134–1168, 2006.

[3] Wangchao Le, Feifei Li, Anastasios Kementsietsidis, and Songyun Duan,"Scalable Keyword Search on Large RDF Data" IEEE transactions on knowledge and data engineering, vol 26, no 11, pp 2774-2788, 2014
[4] Aruna M.Gunje*, Dr.Suhas Raut**,"Designing approach for RDF and OWL files using dynamic query forms " Indian Streams Research Journal , Volume - 5| Issue - 7 | Aug - 2015 ISSN 2230-7850
[5] Q. T. Tran, C.-Y. Chan and S. Parthasarathy. "Query by output". In Proceedings of SIGMOD, pages 535–548, Providence, Rhode Island, USA, September 2009.
[6] C. Li, N. Yan, S. B. Roy, L. Lisham, and G. Das, "Facetedpedia: Dynamic generation of query-dependent faceted interfaces for wikipedia," in Proc. WWW, Raleigh, NC, USA, Apr. 2010, pp. 651–660.
[7] K. Chen, H. Chen, N. Conway, J. M. Hellerstein, and T. S. Parikh, "Usher: Improving data quality with dynamic forms," in Proc. ICDE, Long Beach, CA, USA, Mar. 2010, pp. 321–332.
[8] E.Chu , A. Baid, X. Chai, A. Doan, and J. F. Naughton, "Combining keyword search and forms for ad hoc querying of databases," in Proc. ACM SIGMOD, Providence, RI, USA, Jun.2009, pp. 349–360.
[9] Q. T. Tran, C-Y. Chan and S. Parthasarathy. Static Checking of Dynamically Generated Queries in Database Applications , Rhode Island, USA, September 2009.
[10] S. B. Roy, H. Wang, U. Nambiar, G. Das and M. K. Mohsnia. Dynacet: Building faceted search systems over databases. In proceedings of ICDE Conference, pages 1463-1466, Shanghai, China, March 2009.
[11] M. Jayapandian and H. V. Jagadish. Automating the design and construction of query forms. IEEE TKDE, 21(10): 1389- 1402, 2009.
[12] S. B. Roy, H. Wang, U. Nambiar, G. Das and M. K. Mohsnia. Dynacet: Building faceted search systems over databases. In proceedings of ICDE Conference, pages 1463-1466, Shanghai, China, March 2009.
[13] G. Chatzopoulou, M. Eirinaki and N. Polyzotis. Query recommendations for interactive database exploration. In proceedings of SSDBM, pages 3-18, New Orleans, LA, USA, June 2009.
[14] T.Tran , H. Wang, S. Rudolph, and P. Cimiano, "Top-k Exploration of Query Candidates for Efficient Keyword Search on Graph- Shaped (RDF) Data," Proc. IEEE Int'l Conf. Data Eng. (ICDE), 2009.