An Enhanced Approach for Querying Integrated Web Analytics Ontology Using Quepy

Reshma K[1*], Sindhu S[2]

[1]PG Scholar, Department Of Computer Science and Engineering, NSS College of Engineering, Palakkad
[2]Associate Professor, Department Of Computer Science and Engineering, NSS College of Engineering, Palakkad

Abstract: ***Web analytics is becoming an inevitable aspect in today's Ecommerce scenario, It allows the companies and Ecommerce website owners to track the behavior of customers visiting their website which helps them to improve their website. There exists a series of tools for web analytics in market such as google analytics, piwik etc. .But most of the tools focus on low level and limited set of attributes. So here comes the relevance of an ontology based approach to collect web analytics data from many sources of web analytics tools, as a result a web analytics ontology(WAO) is created using protégé tool. This WAO considers large and complimented set of web metrics and attributes. Searching this web analytics ontology is done by means of SPARQL query. And that is a keyword based search which searches enormous amount of data. User takes more time to access relevant information. And also it is quite difficult for a beginner to learn such a high level Query language So, in order to overcome the limitation of using such a high level query language for accessing the web analytics ontology .i.e. searching by meaning instead of search by literal string. Search engine interpret the meaning of user's query and the relations among the concepts that the ontology contains with respect to particular domain. In this paper, we proposed querying system in which, user enters a natural language query and for that query meaningful concepts are Extracted from domain ontology, For all the terms (expanded and initial query terms), SPARQL query is formed and then it is fired on the knowledge base(ontology) that finds appropriate RDF triples in knowledge Base. In our proposed system we are converting the natural query to SPARQL query using Quepy tool and creating the web analytics ontology by using Protégé tool.***

Keywords: Web analytics, ontology, RDF, Data Integration, SPARQL, Quepy, NLTK

1. Introduction

Web analytics can be used to track where your web traffic is coming from, what types of products the users are interested in, what kinds of keywords people are typing into search engines to arrive at your website etc. it can be also used to track where and how one's traffic converts to sales leads, and where those leads come from. This kind of information is inevitable in helping one to improvise your online marketing and sales generation leads, and improving user experience further. It also helps to know in detail about which phrases to focus on for your ongoing search engine optimization efforts. Ignoring web analytics is just like throwing away opportunities to improve not only the functionality of your website, but also to expand, focus or develop your product offerings to best meet the needs or your target market.

Web analytics plays an important role in today's Ecommerce world. Presently used web analytics tools such as Google analytics, clicky, piwik etc focuses on low level attributes an limited set of features. But this is not sufficient for a systematic analytics purpose. As a result Ontology based data integration approach for web analytics[1] is proposed. Which collects web analytics data from many sources of commercial and digital footprints, and web analytics ontology is created called wao.owl.

Thomas Gruber, defines ontology as "explicit specification of conceptualization", means that ontology describe a relationship between different semantic concept and their properties. Ontology has a vital role in access and interchange of information, use and reuse of knowledge, sharing of information, and common understanding of specific domain are communicated among people for developing their applications .Ontology based semantic search means ability to access the most relevant data for user query from our knowledge base. Here the web analytics ontology has a total of 62 classes (groups of individuals sharing the same attributes), 61 object properties (binary relationships between individuals), and 67 data properties (individual attributes), 33 restriction axioms and 3 individuals[2].so using this web analytics ontology a cutting edge analytics can be done, which will be far more accurate than using individual web analytics tool. Here data can be queried by using a high level query language called SPARQL. Here user is provided with more results, from that finding the relevant data and learning such a high level query language for accessing web analytics data is challenging task. Here in this work Web analytics ontology describe things and their properties and interrelations related to web analytics in a way that computers can process and automate.. In our proposed system, web analytics ontological database is in RDF data format. For RDF data format SPARQL is a query language since this SPARQL Query has the following limitations:

- Difficult to handle by a naive user/beginner
- The maximum number of rows that will be returned is 100, 000.
- There is a maximum query execution time of 10 minutes.

So to make the process of querying the web analytics ontology much more virtuous, we are converting the natural input query to SPARQL query. After that SPARQL query is fired on RDF to retrieve adequate data.

2. Related Works

2.1 A framework for semantics preserving SQL-to-SPARQL translation

There have been several attempts to make RDBMS and RDF stores interoperate. The most popular one, D2RQ[2], has explored one direction i.e. to look at RDBMS through RDF lenses. This approach present RETRO, which explores the reverse direction i.e. to look at RDF through Relational lenses. RETRO generates a relational schema from an RDF store, enabling a user to query RDF data using SQL[8]. A remarkable advantage of this direction in addition to interoperability is that it makes numerous relational tools developed over past several decades, available to the RDF stores. In order to signalize interoperability between these two DB systems one needs to resolve the heterogeneity between their respective data models and include schema mapping, data mapping and query mapping in the transformation process. However, like D2RQ, RETRO chooses not to physically transform the data and deals only with schema mapping and query mapping.

RETRO's schema mapping derives a precise area specific relational schema from RDF data and its query mapping transforms an SQL query over the schema into a provably equivalent SPARQL query [8], which in turn is executed upon the RDF store. Since RETRO is a read only hypothesis, its query mapping uses only a relevant and relationally complete subset of SQL. A proof of correctness of this sublimation is given based on compositional semantics of the two query languages. The interoperability between RDF Stores and RDBMS by firstly deriving a relational schema from the RDF store and secondly providing a means to translate an SQL query to semantically equivalent SPARQL query. The future versions of RETRO[9] plan to provide an algorithm for deriving a user friendly relational schema and extend the query mapping algorithm with additional and advanced SQL operators such as aggregation. Another promising direction for the future work is to leverage the query optimization available for SQL queries to generate efficient, equivalent SPARQL queries.

2.2 SQL to SPARQL mapping for RDF querying based on a new efficient schema

Here an algorithm for querying RDF data using SQL without conversion of RDF instances is introduced. This algorithm translates an SQL query into an equivalent SPARQL query that is to be directly executed on the RDF data and allows it for SQL users to expeditiously and easily query the RDF data. The SQL queries are formulated based on the converted relational database schema that the algorithm builds from the RDF one. In this algorithm not only simple SQL queries are considered but also complex ones such as those with UNION, INTERSECT or EXCEPT expressions [9].Resource Description Framework has been standardized by the W3C as the language of the semantic web to reflect the semantics of the data being exchanged on the web. It comes with an emerging data format that makes it possible to share the meaning of data between various applications. However because of the dominance of relational database systems and associated tools that are still based on SQL for handling data there is an increasing need for tools to help SQL users to query RDF data. In this perspective this work specifies an approach for querying RDF data using SQL. The technique used is based on modeling RDF data by a suitable relational schema that makes it possible for users to query RDF data with SQL without any instance translation into relational tables. Based on the extracted relational schema our approach converts users SQL queries into equivalent SPARQL queries to be executed on the RDF data. This is done in efficient way since the proposed modeling technique insures a consistent representation of all information in RDF data that avoids redundancy and comes up with a minimal set of representation tables in the extracted schema[11]. Because of this concise modeling technique we aim to further use it in the future for integration with existing relational database systems for purposes related to RDF data storage and manipulating. This will open a new era for existing relational systems to be open for extensions to the world of semantic web.

All these related works deals about mechanisms to convert an SQL Query to SPARQL Query, but here this work introduces an entirely different concept of converting an NL Query to SPARQL Query which can be implemented in this web analytics concept. It can be accomplished by a python framework called Quepy which converts NL Questions to SPARQL Query, But Quepy works basically on DBpedia or freebase database. So by altering the access database of Quepy to web analytics database this can be achieved.

3. Querying Integrated web analytics ontology using Quepy

Here the domain where data retrieval is focusing into is Web analytics ontology. This ontology considers large and complimented set of web attributes so that the data retrieved will be far more accurate. This domain ontology is created using protégé tool.Protégé is a tool which convert data into RDF data format. We have used Protégé tool to create web analytic ontology .This ontology is for storing information about of various classes, individuals and properties with their attributes and relationships related to web analytics.Figure 1 shows the creation of attributes of web analytics class and their subclassess. The general overview of the web analytics ontology created is shown in figure2.

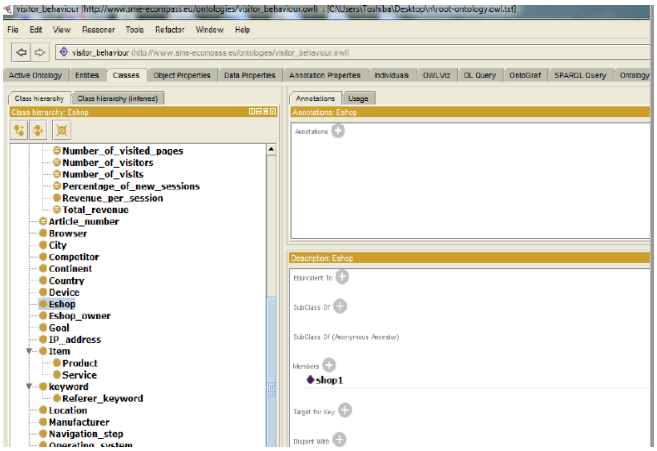


Figure 1: Creation of Eshop class by using protégé tool

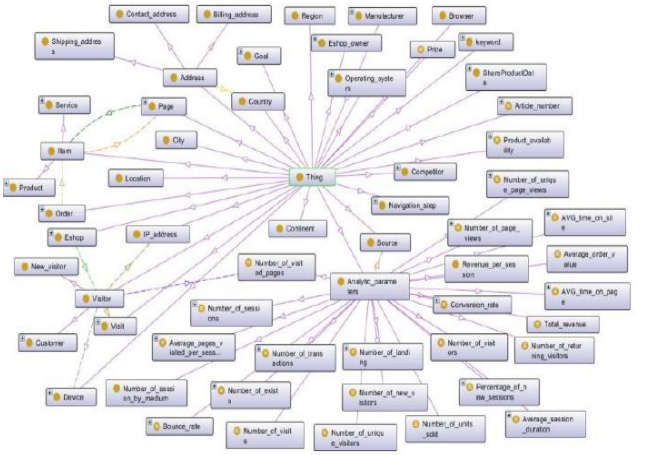


Figure2: Ontograph representation of web analytics ontology

The retrieval of web analytics data is done by using SPARQL Query. Since this ontology considers a large set of attributes and metrics the retrieved data will gives a precise web analytics report..But the querying mechanism used here is SPARQL which has several drawbacks. So we developed a natural language querying interface for retrieving data from this web analytics ontology. This can be implemented by using a python framework called Quepy.

3.1 Quepy Framework

Quepy is a framework for transforming natural language questions into database queries. It is developed by a company called Machinalis. The query transformation is done by first applying a regular expression method, where the regular expressions are predefined to match the natural language questions. The input question is parsed using a library called REfO (Regular Expressions of Objects), which is also developed by Machinalis. That uses tokenization and lemmatization on the words. After this the system uses the NLTK interface to part-of-speech tag the tokens in the input question.

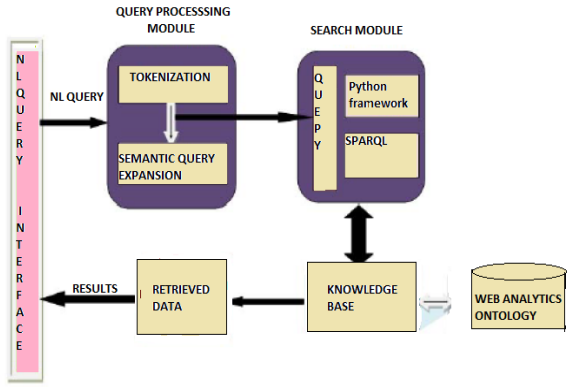
Figure 3: Quepy querying interface

Natural Language Toolkit (NLTK) is an platform for Python that handles several natural language processing issues, in this case part-of speech tagging is used. Quepy then constructs the tokens into triples in adjacency lists, merges the triples, and prints them out as a SPARQL-query. Quepy querying interface is given in figure 3.The Quepy project is a commercially built natural language to query interface, it has a working test demo online where one can translate questions about counties and movie-stars into queries. This is a system that works perfectly within a controlled sentence structure. Quepy framework works on dbpedia and freebase data base. So by altering this quepy interface to deal with the proposed web analytics ontology an advanced Natural language querying interface for querying webanalytics data can be implemented

3.2 System Architecture

We proposed a framework for querying the data from web analytics ontology in triple format by means of a natural language query. We have presented a ontology based technique for query expansion that has been integrated with the help of specified tools, to our search scheme. To sum up, the querying process includes certain steps:

The Query conversion mechanism converts user input query into SPARQL query .The SPARQL is a language for RDF database .Thus generated SPARQL query is mapped on the knowledge base, which has been stored in RDF format. SPARQL query find the semantic relation in ontology. Semantic information is retrieved to user. We have presented the overall architecture for our system shown as in Figure.4. The system consists of four basic modules: User Interface module, Query processing module and Search Module & Ontology construction module. They are explained below.



.**Figure 4**: Proposed system Architecture

A. Domain Ontology Construction Module

Firstly, detailed information of the domain from various sources is gathered, here the domain is web analytics. The attributes are Eshop, customer etc Identification and setting of classes and subclasses for the ontology to be developed is done at second stage. Identification and setting of object and data properties between classes and subclasses is done at third stage whereas their domain and range is set at fourth stage. Comments for domain explanation are added to the classes and properties. Creation of class instances and setting their properties (both data and object) is the fifth stage. Consistency check is performed at sixth stage for which various inbuilt reasons like HermiT can be used. Seventh stages to save the ontology in RDF/OWL format. In last, ontology is exported in RDF/OWL format for execution of queries at the desired interface (Bansal and Chawla, 2014). Finally, the prototype ontology is developed for web analytics domain having more than 350 RDF triples Ontology can be created for a particular domain by using protégé tool and here the integrated web analytics ontology is also created by the same tool which consists of large set of attributes and concepts (various terms of a specific domain), relationships between concepts. Ontology show the hierarchical relationship between different classes and their subclasses in graphical pattern as shown in figure 2:

B. User Interface Module

The user can give input query that they wishes to retrieve from the integrated web analytics ontology. An example of such a query can be "Which product has high number of purchase rate?" After the processing of other modules the user will get the relevant information accordingly from the web analytics ontology.

C. Query Processing Module

The query given by the user through user interface is handled by query processing module The meaningful concepts are extracted through the tokenization of the input query and expanded through semantic query expansion by using word net[15].WorldNet is used for find the synonyms of the words. The natural language question given by the user is then semantically expanded to SPARQL query language by using a quepy tool. The quepy convert query to sparql semantically by using NLTK_DATA contains WordNet. .

D. Search Module

This module is able to interpreting the user's input query.
In this module input query is converted into sparql query which find the semantic RDF data in domain web analytics ontology. SPARQL is a query language for RDF data. It works like an SQL query language for Relational data [10]. But the main difference is that SQL accessing the data stored in Tables and SPARQL accessing the data in given namespace/ontology. Namespace is also called prefix which is in the URI link, Means that accessing the data from that resources. So, SPARQL is more advanced than SQL because SPARQL access relational as well as diverse data stored in link. SPARQL is a RDF query language for semantically map on a RDF data source and retrieve semantic data from Knowledge base. SPARQL show the semantic meaning of user query for retrieve semantic data from RDF. SPARQL match the prefix i.e. namespace with RDF namespace then search data from those URL properties. Consider the example"When was Ravi Menon appointed in HSBC? SPARQL Query generated is shown in figure 5 below:

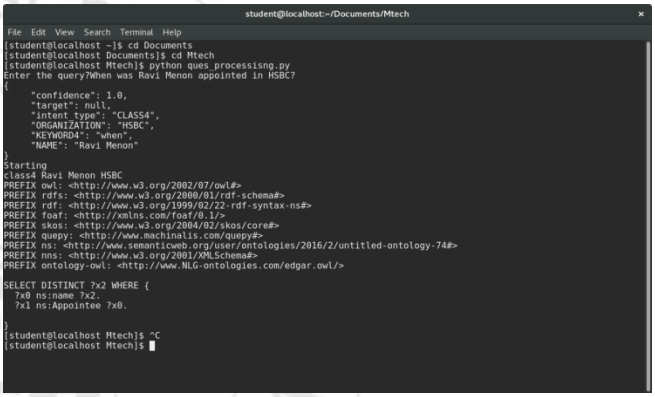


Figure 5: SPARQL Query generator

4. Conclusions

We have proposed the approach of querying the web analytics ontology by means of Natural Language query instead of SPARQL Query. This approach for retrieving web analytics ontology system solves the limitation of system using the high level query language named SPARQL. Sparql query is used to extract the RDF data with respect to user input query .Quepy a python framework is used for converting natural query to sparql query. Protégé tool is used for creating ontology in RDF data format.

The most fundamental step is ontology development which is shown through a prototype developed in web analytics domain This system overcomes the limitation of learning such a high level query language and thereby making querying simple and accurate. Experimental result shows that our system retrieve some type of queries answered. Future work will extend as Querying system which gives the answer of all possible types of questions and also question suggestions.. This work can be enhanced on real time data which will extract information at runtime after the user's query is processed and understood.

References

[1] Akanbi, A. K and F.Christy .(2014) ”Lb2co: A semantic ontology framework for b2c ecommerce transaction on the internet”, International Journal of Research in Computer Science, volume 4, pp. 1-9.

[2] Maria del Mar Roldan Garcia, JoseGarcia-Nieto and JoseF. Aldana-Montes(2016)”An ontology-based data integration approach for WA in e-commerce”, Expert Systems with Applications, vol 6, pp. 20-34.

[3] Hepp M and F.Deborah .(2008) ”Goodrelations: an ontology for describing products and services offers on the web” In Proceedings of the 16th international conference on knowledge engineering and knowledge management, pp. 332–347.

[4] Tamma.V, Phelps.S. (2005), Dickinson.I and Wooldridge.M ”Ontologies for supporting negotiation in e-commerce.” Engineering Applications of Artificial Intelligence, vol 7, pp.223-236.

[5] Waralakv S and McGuinness (2008) ”Learning semantic web from e-tourism.” Agent and multi-agent systems: Technologies and applications, In Lecture Notes in Computer Science, pp.516-525.

[6] Trastour, Bartolini C and Preist C (2003) ”Semantic web supportfor the business–to business e-commerce pre-contractual lifecycle.”, Computer Networks, vol 6, pp.661 -673.

[7] Natalya N, McGuinness and F.Deborah (2010) ”Ontology Development 101: A Guide to Creating Your First Ontology”, Technical Report, tanford University Knowledge Systems Laboratory Technical Report, pp 632-643.

[8] Jyothsna Rachapalli, Vaibhav Khadilkar, and Murat Kantarcioglu and (2013) ”RETRO: A Framework for Semantics Preserving SQL-to-SPARQL Translation”, The International jounal on Computational Science, vol 5, pp 180–191.

[9] Svetlana Chuprina, Igor Postanogov and Olfa Nasraoui (2015) ”Ontology Based Data Access Methods to Teach Students to Transform Traditional Information Systems and Simplify Decision Making Process”, The International Conference on Computational Science, vol 80, pp 1801–1811.

[10] Christopher J, Prom S (2011) ”Using Web Analytics to Improve Online Access to Archival Resources”, The American Archivist, vol 6, pp 32-43.

[11] Okazaki S and Rivas J (2011) ”Analysing The Impact of visitors on page views with google analytics”, International Journal of Web and Semantic Technology, vol 2, pp 1-19.

[12] Calvanese D, De Giacomo and LemboD, Lenzerini MandRosati R (2009) ”Conceptual Modeling for Data Integration”, Conceptual Modeling: Foundations and Applications, vol 4, pp 173–197.

[13] Chuprina S and Nasraoui O(2015) ”Using Ontology-based Adaptable Scientific Visualization and Cognitive Graphics Tools to Transform Traditional Information Systems into Intelligent Systems”, Scientific Visualization, vol 8, pp 23-34.

[14] Douglas D, LePendu Pand Kim S (2012)”Integrating Databases into the Semantic Web through an Ontology-Based Framework”, Proceedings of the 22nd International Conference on Data Engineering Workshops (ICDEW’06).

[15] Natalya N, McGuinness and Deborah L (2010) ”Ontology Development 101: A Guide to Creating Your First Ontology”, Technical Report, tanford University Knowledge Systems Laboratory Technical Report, vol 6, pp 632-643, 2008