

HOTEL MANAGEMENT SYSTEM an E-COMMERCE APPLICATION using RESOURCE DESCRIPTION FRAMEWORK (RDF)

A. Gauthami Latha¹, R. Naga Raju¹, Ch. Satyanarayana³, Y. Srinivas⁴

¹Department of Computer Science and Engineering, V.I.I.T, Duvvada, Visakhapatnam, A.P, INDIA

²Department of Computer Science and Engineering, V.I.I.T, Duvvada, Visakhapatnam, A.P, INDIA

³Department of Computer Science and Engineering, J.N.T.U.K., Kakinada, A.P, INDIA

⁴Department of Information Technology, GITAM University, Visakhapatnam, A.P, INDIA

Abstract—The Semantic Web is an intelligent retrieval system, where the information is associated in a way that can be processed easily by machines, worldwide. At present, only human are able to understand the product information that is available online. The emerging semantic web technologies have the potential to extremely influence the further development of the Internet market. This paper provides a review and the background of Resource Description Framework (RDF), an web ontological language. This project specifically explains the Hotel Management based on the ontology language RDF. In this project we have applied the technique of Sparql Protocol and RDF Query Language [SPARQL] on RDF for semantic web mining on non RDF database. RDF is used to generate the customer details for the Hotel, JENA API is used to extract the order details, feedback from the customers and at the administrator side SPARQL query language is used, to retrieve the information depending on the query passed. Screen shots of the hotel management application were presented using Net Beans IDE.

Keywords - Hotel Management, Jena, Net Beans IDE, Ontology, RDF, Semantic Web, Semantic Web Mining, SPARQL.

I. INTRODUCTION

In current technology, storage and retrieval of information from relational databases is done by using Structured Query Language (SQL), but the main disadvantage with the use of SQL, is that the accessing speed is low, and the load on the server is more. It does not work efficiently for large data bases and when the nested query is passed. The semantic web mining helps to overcome the limitations by using SPARQL on RDF.[7]. RDF is a frame work used to store the data and SPARQL is used to retrieve the information from RDF using Jena API.

A. Limitations of Current Web

In present world most of the web applications are based on relational databases to access the information, but it fails to gain the relevant information efficiently within short span of time [14, 15].

An illustration of the query possessing mechanism in SQL is given below. Let us consider an example, where we are interested in finding the population in a particular City of USA, particular area of interest, and transportation costs to travel, in order to determine the relationship between population density and public transportation costs, the mechanism adopted by SQL in retrieving the query are

- 1) To get information regarding the city, it takes the help from Wikipedia.
- 2) To retrieve the data with respect to transportation cost, it takes the help from another source

- 3) And to extract the data regarding population, area and transportation cost, it combines the queries 1 and 2 which is a nested query. [14].

To access the relational database SQL needs nested queries, this increases the load on the server, there by the accessing time becomes high and finally results in the degradation of the performance. All these limitations can be overcome by using SPARQL.

The advantages of SPARQL are highlighted in this paper by considering a Hotel Management system as an example, which mainly focuses on the items available in the Hotel, the customer details and his feedback [1, 2]. In section II related terminologies are discussed. Section III introduces the architecture of hotel-management model, In Section IV case study is explained and the results obtained are summarized in Section V. Finally the conclusion is given in section VI.

II. RELATED TERMINOLOGY

A. Ontology

Ontology plays main role in searching the relevant documents in the semantic web, more over it helps in getting the accurate information. Ontology is the heart of the semantic web applications and it is metadata representation. Ontology consists of concepts (also known as classes), relations (properties), instances and axioms.

Ontology is defined as a 4-tuple {C, R, I, Ai}, where C is a set of concepts, R is the set of relations, I, the set of instances and A, the set of axioms [5]. Several tools are available in literature, namely Altova, semantics, protégé, ontolingna, UNSPSC, Rosetta Net to generate the ontology automatically and Jena API for generating the ontology manually [10, 16]. OWL, FOAF, RDF is some of the ontology languages [15]. In the present paper, RDF is mainly focused.

1) RDF (Resource Description Framework)

RDF is a W3C standard used for describing resources using a simple model based on named relationships between resources. RDF is a Meta language based on XML (Extensible markup language).RDF is a composition of subject, predicate and object in which a subject has a property whose value can be either a string, literal or a reference to another resource[3,4,6]. It is particularly good for storing metadata about files and other machine-accessible resources. [12]

Consider the Customer with name is keeru and Mobile number of the Customer is 9392514180.

Here, the **subject** is Customer. The **predicates** can be either name or mobile number. And the **objects** can be keeru or 9392514180

The syntactic representation is as follows:

- < Subject, predicate, object>
- < Customer, Name, Keeru> or
- <Customer, Mobile Number, 9392514180>

The graphical presentation for the above example is shown in Fig. 1.

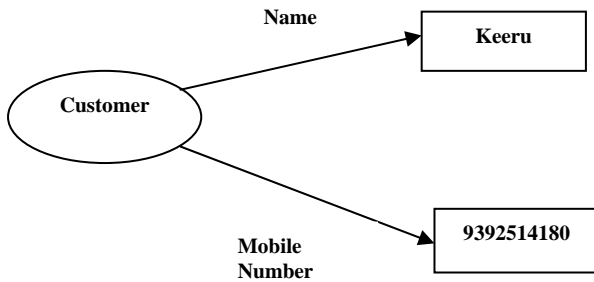


Fig. 1 RDF Graph Example

B. Applications of RDF

- 1) *Resource Discovery*: RDF is used to discover the resources on the Web more easily.
- 2) *Cataloguing*: RDF will enable users to better describe the data and relationships available at a particular Web page, site or digital library.
- 3) *Content Rating*: RDF will allow content to be rated.
- 4) *Digital Signatures*: In E-Commerce, Collaboration, and other applications RDF will be a key to develop the “Web of Trust”.

C. Jena API

Jena is a Java framework for writing Semantic Web Applications. The Jena ontology API is intended to support programmers working with ontology data based on RDF. Specifically, this means support for OWL, DAML+OIL and RDFS [8, 15]. A set of Java abstractions extend the generic RDF Resource and Property classes to model more directly the class and property expressions found in ontology using these languages, and the relationships between these classes and properties, and the individuals created from them[9].

1) Conversion to RDF using Jena API:

Let us consider the following example - Customer Name - Keeru and City- Vizag. Table I shows the java program and the table II shows the related RDF that is generated for the example.

```

public class x extends Object {
public static void main (String args[]){
Resource node;
Model m = ModelFactory.createDefaultModel ();
String users = "http://somewhere/users";
String username = "keeru";String city = "vizag";
Property Username = m.createProperty (users + "Username");
Property City = m.createProperty (users + "City");
node=
m.createResource("http://somewhere/users").addProperty(User
name, username).add Property(City, city);
m.setNsPrefix ("users", users);m.write (System.out);}}
  
```

TABLE I. PROGRAM in JAVA

```

<rdf:RDF
xmlns:users="http://somewhere/users"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-
ns#"
xmlns:users="http://somewhere/" >
<rdf:Description rdf:about="http://somewhere/users">
<users:City>vizag</users:City>
<users:Username>keeru</users:Username>
</rdf:Description> </rdf:RDF>
  
```

TABLE II. RDF GENERATED

D. SPARQL Query Language

TABLE III. SQL AND SPARQL

Features	SQL	SPARQL
Data Storage	Non RDF/ relational database	RDF/OWL/ FOAF
Data Access	Based on syntax	Based on ontology
Queries	Nested/multiple queries	Single query
Load on Server	More in accessing relational data	Less in accessing relational data
Accessing Speed	Less	Comparatively high

SPARQL is an RDF query language and protocol developed within W3C. SPARQL defines a standard query language and data access protocol used for RDF data model [3, 12]. SPARQL works for any data source that can be mapped to RDF. Jena provides the ARQ query engine which is a complete implementation of the SPARQL query language [4]. SQL is for relational database whereas SPARQL is for Semantic Web.

In section I.A the limitations of the current web are presented. SPARQL overcomes the limitations of SQL. Using SPARQL, the application can be accomplished by a single query that merges the appropriate data source. The comparisons between SQL and SPARQL are presented in Table III.

III. ARCHITECTURE

A customer should first register himself/herself, once the registration process is completed, he gains the privileges to order the items and also to express his feedback. Once the customer provides the information, the non- RDF database is updated and converted to RDF simultaneously by using Jena API. Administrator will know the customer feedback regarding the hospitality provided by the management by .passing query using SPARQL through HTML browsers. The SPARQL maps to RDF file, retrieves the results and display it in the HTML browser. Administrator will know the customer information to develops their business through some fields which include

- 1) Id- Customer identify
- 2) Feedback type

Through customer id, administrator will get the particulars regarding the customer, feedback type enables to know the identity of the customer who has given a particular feedback

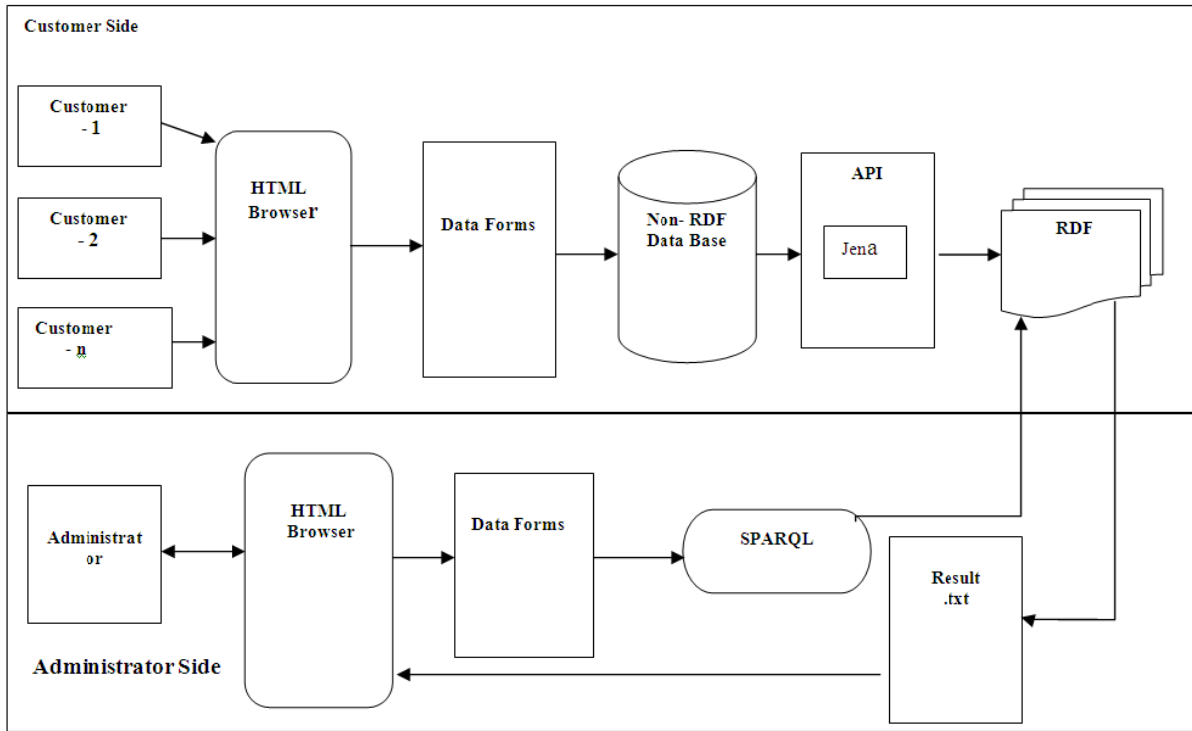


Fig. 2 Architecture of Hotel Management System

In this case the ontology agent will create the RDF for Registration & Feedback simultaneously. Using Jena the RDF generated and is shown in the Fig. 2.

IV. CASE STUDY

A. Customer Side

1) Customer/User Registration & Feedback Forms:

Let us consider the Registration Form and Feedback Form, the GUI Forms for Registration and Feedback is as shown in the Fig. 3 and Fig. 4 with the help of screen shots.

The registration form is divided into two sections:

- Access Information:** Includes fields for Username (keeru), Password, Confirm Password, Hint Question (Whats my fathers name), and Answer (vissu).
- More Information:** Includes fields for Full Name (keerthana), Gender (female), D.No. (90gF), Street (vnagar), City (bhimavaram), State (Andhra Pradesh), Country (India), Email Id (keerthi.n@gmail.com), and Mobile Numer (9392514180).

Fig. 3 Screen shot of Customer Registration form

The feedback form includes a rating section and a table for service evaluation.

Overall how would you rate your stay?
 Exceeded expectations Met expectations Did not meet expectations

The reason you have chosen us
 Please rate the following by importance (6=most important; 1=least important)

Price: 4 Location: 4 Staff performance: 4 Quality: 4

Type	Did Not Meet expectations	Met Expectations	Exceeded Expectations	No Comment
Reservations	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
Reception (Check In)	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Promptness of Wake up Call	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
Security	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
Reception (Check Out)	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>

Fig. 4 Screen shot of Customer Feedback form

2) Customers RDF for Feedback and Registration: In this case the ontology agent will create the RDF for Registration and Feedback simultaneously. Using Jena the RDF is generated and it is represented in the table IV.

TABLE IV. RDF for REGISTRATION & FEEDBACK FORMS

```

<rdf: Description rdf: nodeID="A0">
  <users: Id>2</users: Id>
  <users: Pwd>keerutha</users: Pwd>
  <users: State>Andhra Pradesh</users: State>
  <users: Feedbacktype>complaint</users: Feedbacktype>
  <users: Comments>not bad</users: Comments>
  <users: Street>vnagar</users: Street>
  <users: Cleanliness>2</users: Cleanliness>
  <users: Fid>4</users: Fid>
  <users: Fname>keerthana</users: Fname>
  <users: Internetaccess>2</users: Internetaccess>
  <users: Security>3</users: Security>
  <users: Mobile>9392514180</users: Mobile>
  <users: Sex>female</users: Sex>
  <users: Address>30gf</users: Address>
  <users: Username>keeru</users: Username>
  <users: Country>India</users: Country>
  <users:Email>keerthi.n@gmail.com</users:Email>
  <users: Mobile>9392514180</ users:Mobile>
  <users: City>bhimavaram</users: City>
  <users: Roomservice>2</users: Roomservice>
  <users: Expectations>Exceeded expectations
  <users: Expectations>
  < users: comments > not bad</users: comments>
  < users: Email>keerthi.n@gmail.com</ users: Email>
  <users: Arrivaldate>07/07/2011</users: Arrivaldate>
  </rdf: Description>
    
```

The RDF in table 4 describes the features of both the Registration and Feedback Forms of the single Customer/User. In the same way the RDF can also be generated for multiple users at a time. Section II.A.I presented the syntax of RDF, the Registration and Feedback RDF also represents their features in the format – <Subject, predicate, object>. Like <users: Username>keeru</users: Username> <users:Comments>notbad</users:Comments> <users:Email>keerthi.n@gmail.com</users:Email>

B. Administrator Side

1) Case 1. Available Customer/User Details:

At the administrator side, when the administrator wants to know the details of the customers registered, can be viewed as shown in the following Fig. 5 in the form of screen shot. The SPARQL- query that is passed to view the customer details are represented as shown in the table V.

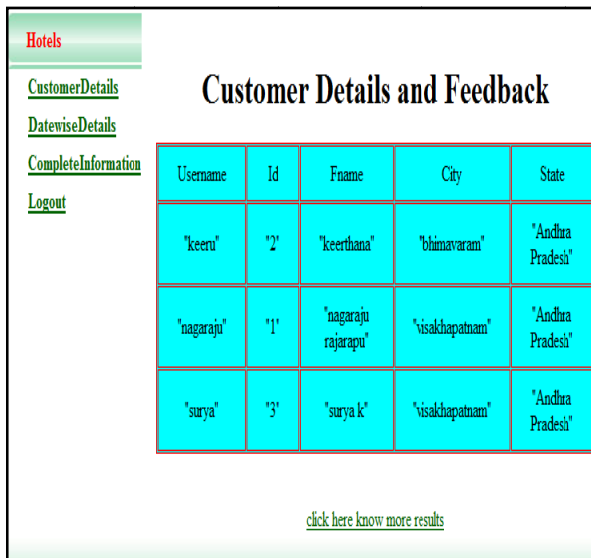


Fig. 5 Screen shot of available customer

TABLE V. QUERY to RETRIEVE AVAILABLE CUSTOMERS

```

PREFIX users: <http://fakestd/users#>
SELECT DISTINCT ?Username ?Id ?Fname
?City ?State
FROM <http://fakestd/users#>
WHERE {
  ?element users: Username ?Username .
  ?element users: Id ?Id . ?element users: Fname
  ?Fname .
  ?element users :City ?City .?element users: State
  ?State . }
    
```

2). Case2: Customer/ User Id:

In this case, if the administrator wants to know the particular details of the single customer and the food menu selected by a customer, he can have the details as in the following Fig. 6 and Fig. 7. The related query is represented in the table VI. The final result is as shown in the Fig. 8, after executing the SPARQL[4] query related to the customer ID.



Fig.6 Screen shot to ask Customer Information

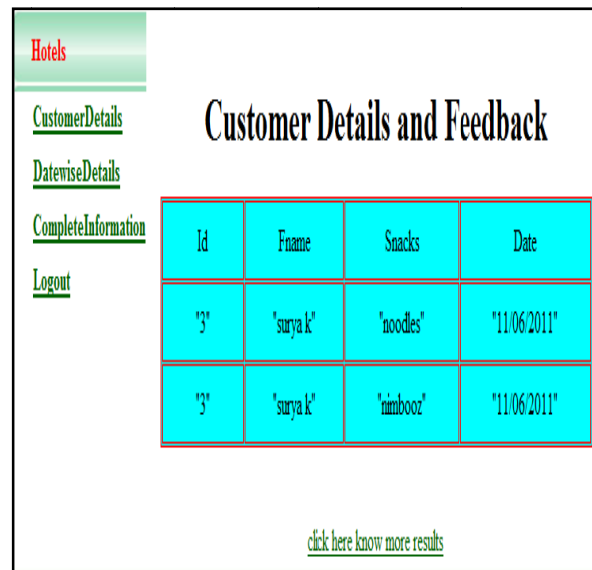


Fig. 7 Screen shot for customer information with ID "3"

```

PREFIX users: <http://fakestd/users#>

SELECT DISTINCT ?Id ?Fname ?Snacks ?Date

FROM <http://fakestd/users#>

WHERE { ?element users: Id "\"+Id+"\". ?element
users: Id ?Id. ?element users: Fname ?Fname.
?element users: Snacks ?Snacks.
?element users: Date ?Date.}
    
```

TABLE VI. QUERY to ACCESS CUSTOMER INFORMATION with ID "3"

```

PREFIX users :< http://fakestd/users#>
SELECT DISTINCT ?Id ?Username ?City
?Snacks ?Date ?Fid ?Mobile ?Email ?Comments
?Expectations
FROM <http://fakestd/users#>
WHERE { ?element users:Expectations "\"+
Expectations + "\" . ?element users:Id ?Id.?element
users: Username? Username?element users: City
?City.?element users: Snacks? Snacks?element
users: Date? Date.?element users: Fid
?Fid.?element users: Mobile ?Mobile.?element
users:Email ?Email element users:Comments?
Comments?element users:
Expectations? Expectations.}
    
```

TABLE VII. QUERY to ACCESS CUSTOMER DETAILS BASED on FEEDBACK

Case 3) Customer/User Feedback

In this case, if the administrator wants to know the details of the customers, food items and feedback form through feedback type. That has been shown as in the following Fig. 8 and Fig. 9.

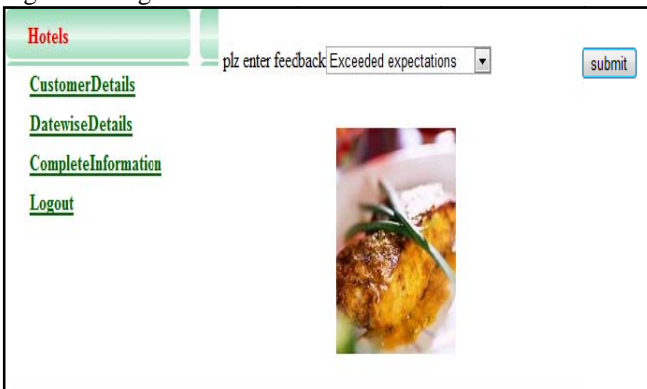


Fig. 8 Screen shot for customer feedback type.

Id	Username	City	Snacks	Date	Fid	Mobile	Email	Comments	Expectations
"2"	"keeru"	"bhimavaram"	"burger"	"05/05/2011"	"4"	"9392514180"	"keertlin@gmail.com"	"not bad"	"Exceeded expectations"
"2"	"keeru"	"bhimavaram"	"fishry"	"05/05/2011"	"4"	"9392514180"	"keertlin@gmail.com"	"not bad"	"Exceeded expectations"
"2"	"keeru"	"bhimavaram"	"maazza"	"05/05/2011"	"4"	"9392514180"	"keertlin@gmail.com"	"not bad"	"Exceeded expectations"

Fig. 9 Screen shot for results of customers with feedback" EXCEEDED EXPECTATIONS"

Table VII give complete details of the customers based on the query passed. In this the query is generated based on the feedback type "Exceeded expectation". The final result is generated based on the query that is as shown in the figure 9 and the corresponding SPARQL Query that is passed is as Shown in the table VII.

V. RESULTS

The results obtained from the paper are tabulated in the form of graphs.

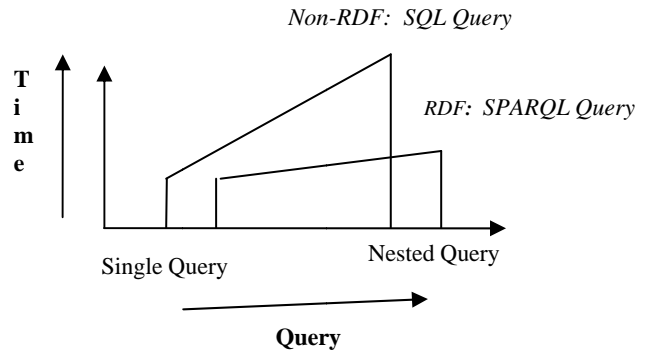


Fig. 10 Query and Time Graph

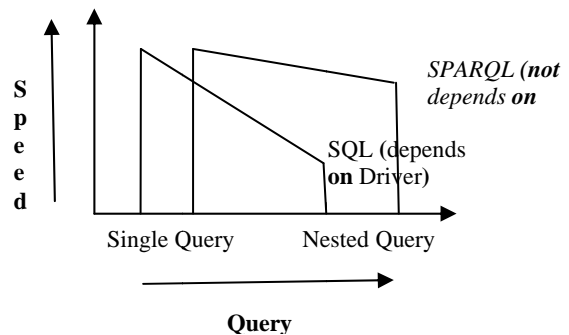


Fig. 11 Query and Speed Graph

The Fig. 10 shows the performance of SQL and SPARQL on Non- RDF database and RDF database respectively. Based on the query passed, i.e. either single query or nested query with respect to the time taken by each database.

SQL is a driver dependent Query language, so initially it can be seen that the query retrieval speed using SQL is faster as the number of nested queries increases, the load on driver increases by this means relational speed will be effected. Where as in SPARQL, it is independent of drivers and hence optimal relation can be achieved as shown in the Fig. 11.

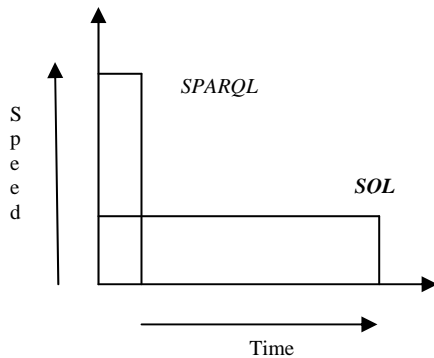


Fig. 12 Time and Speed Graph

Figure 12, shows that in less time, the accessing speed of data in SPARQL is high compared to SQL, as SQL is time consuming to retrieve the data within limited speed.

VI. CONCLUSION

This paper introduces the reader to the concept, applications and background of web mining in particular to hotel management. Screen shots of hotel customer details and feedback are provided using SPARQL an Ontology Language - an overview of RDF language and its usefulness in analysing web mining data is presented. This paper has shown the usefulness of web mining in extracting key words, link analysis of key words and search queries. Web mining of the hotel customer responses makes it

known to management what hotel services need to be improved.

REFERENCES

- [1] Richard S. Segall, Qingyu Zhang , Mei Cao “ *Web-Based Text Mining of Hotel Customer Comments Using SAS® Text Miner and Megaputer Polyanalyst®* “ <http://www.docstoc.com/docs/43446814/WEB-BASED-TEXT-MINING-OF-HOTEL-CUSTOMER-COMMENTS-USING-SAS-TEXT>
- [2] Richard S. SEGALL, Qingyu ZHANG “ *Web Mining of Hotel Customer Survey Data*” [www.iiisci.org/journal/CV\\$/sci/pdfs/QS911DU.pdf](http://www.iiisci.org/journal/CV$/sci/pdfs/QS911DU.pdf)”
- [3] K. Selcuk Candan, Huan Liu, and Reshma Suvarna “ *Resource Description Framework: Metadata and Its Applications*” www.sigkdd.org/explorations/issue3-1/candan.pdf
- [4] Hafiz Hammad Rubbani “ *Semantic Web Solutions*” [www.itu.dk/~hammad/thesis/Final %20Thesis.pdf](http://www.itu.dk/~hammad/thesis/Final%20Thesis.pdf)
- [5] John Davies, Rudi Studer Paul Warren “ *Semantic Web Technologies Trends and Research in Ontology-based Systems*” http://www.booku.com/Semantic-Web-Technologies/John-Davies/ebook_294748.html
- [6] “ *The RDF Advantages* Page “ [http:// www .w3.org/RDF/advantages.html](http://www.w3.org/RDF/advantages.html)
- [7] Michael K. Bergman “ *Advantages and Myths of RDF*” [http://www.mkbergman.com/wp-content/ themes/ ai3/ files/ 2009 Posts /Advantages _Myths _RDF_ 090422. pdf](http://www.mkbergman.com/wp-content/themes/ai3/files/2009Posts/Advantages_Myths_RDF_090422.pdf), April 22, 2009
- [8] “ *The Semantic Web: An Introduction*” <http://infomesh.net/2001/swintro/>
- [9] “ *Jena – A Semantic Web Framework for Java* “ <http://openjena.org/>
- [10] “ *An Introduction to RDF and the Jena RDF API*” [http://jena.sourceforge.net/ down loads.html](http://jena.sourceforge.net/downloads.html)
- [11] Josef Petrak, Jan Zemaneck, and Vojtech Svatek “ *Case Study on Linked Data and SPARQL Usage for Web Application Development*” [http://jspetrak.name /files/PAPER-znalosti2010.pdf](http://jspetrak.name/files/PAPER-znalosti2010.pdf)
- [12] Pierre- Antoine Champin “ *RDF Tutorial*” [www710.univ- lyon1.fr/~champin /rdf-tutorial/rdf-tutorial.pdf](http://www710.univ-lyon1.fr/~champin/rdf-tutorial/rdf-tutorial.pdf), April 2001,
- [13] Steven Lynden, Isao Kojima, Akiyoshi Matono, and Yusuke Tanimura “ *Adaptive Integration of Distributed Semantic Web Data*” [www.cs.rmit.edu.au/~e76763 /tiger _ref/lkmt10-dnis.pdf](http://www.cs.rmit.edu.au/~e76763/tiger_ref/lkmt10-dnis.pdf)”
- [14] Lee Feigenbaum “ *Why SPARQL*” http://www.thefigtrees.net/lee/blog/2008/01/why_sparql.html
- [15] Peter Mika- “ *Social Networks and the Semantic Web*” [http:// dare.uvu. vu.nl /bitstream /1871/ 13263 /5/ 7915.pdf](http://dare.uvu.nl/bitstream/1871/13263/5/7915.pdf)
- [16] *Altova editor* www.altova.com