Ontology Driven Approach for Effective Decision Making

Ashutosh V. Girase
PG Student
Department of Computer Engineering
SSBT's College of Engineering & Technology,
Bambhori, Jalgaon.

Girish Kumar Patnaik
Professor and Head
Department of Computer Engineering
SSBT's College of Engineering & Technology,
Bambhori, Jalgaon.

Sandip S. Patil
Associate Professor
Department of Computer Engineering
SSBT's College of Engineering & Technology,
Bambhori, Jalgaon.

ABSTRACT

Decision-making is the task of every top management in an organization. Decision maker needs relevant and meaningful information to help in taking decisions. Meaningful information retrieval is a challenge for effective decision-making. Due to lack of domain knowledge, meaningful information remains hidden in the database itself. Decisions made out of irrelevant and meaningless information sometimes lead to irreparable damage to organization and its reputation. To retrieve relevant information it is necessary to have background knowledge about the domain. Background knowledge in the form of ontology is an important source of information. Domain ontology used as a source of domain knowledge which retrieves all the meaningful information from the database to help in taking decision. In proposed approach, ontology is used as domain knowledge. Use of ontology improves the relevancy and meaningfulness of the results in order to get more meaningful information for effective decision making. Experimental evaluation shows that, results obtained by using proposed approach are more precise and relevant than existing non-ontological approach.

Keywords

Ontology, Decision-making, Future Prediction, Domain knowledge, Meaningful information, Background knowledge, Information retrieval, Business intelligence.

1. INTRODUCTION

Information retrieval is the process of retrieving information resources relevant to meet the user need. Knowledge base information retrieval is a process of retrieving relevant and meaningful information as per user need from the resources. A recent study on knowledge-base information retrieval shows as a data is growing massively day by day. Retrieval of meaningful information from such a massive data is a very challenging and time consuming task.

Meaningful information is very much important in decision making and problem solving. To retrieve the meaningful information from the database, background knowledge about the domain is necessary. Practically for a single person to have knowledge about every domain is not possible. Due to this, knowledgeable information remains hidden itself in the database. Hence uncertainty in the effectiveness of decision making is occurred as well as wrong decisions may be taken due to wrong information. This is harmful for the organizations reputation and performance.

To overcome this challenge background knowledge about that domain is a key issue. Domain knowledge in the form of ontology is a ray of hope to solve the problem of background knowledge. Ontology is widely used in data mining techniques as a source of domain knowledge. Ontology is a source of knowledge as a set of concepts within the domain, and the relationship between those concepts.

$$O = (C, R, Hc, P, A) \qquad (1.1)$$

Where, C: Set of Concepts

R: Set of Relations

Hc: Concept Hierarchy

P: Properties or Attributes

A: Axioms

Ontology also used to explore the semantic relationship between the concepts and to represent the background knowledge [1]. Background knowledge plays an important role in retrieval of relevant and meaningful information to meet the need of decision maker to take the decision. Decision making is carried out in every organization to solve the problems by using business intelligence process.

Rest of the paper is organized as follows: Section 2 gives an overview of the related work; Section 3 presents the proposed approach; Section 4 presents the result & discussion and Section 5 concludes the proposed approach.

2. RELATED WORK

Ontology is a popular area of research nowadays. Mainly ontology is used in the area of artificial intelligence. Due to lack of semantics, traditional keyword based technique in data mining limits in finding the relevancy and understanding the user need. Ontology has given a new ray of hope to overcome the challenges of data mining. Use of ontology as a domain knowledge repository found too much promising in the various data mining tasks such as information retrieval, information extraction, classification, clustering, recommender system, link prediction etc. Structure of literature survey as ontology driven techniques is shown in Figure 1.

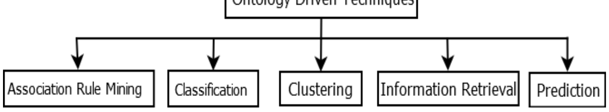

Fig 1: Ontology Driven Techniques

2.1 Association Rule Mining

Xuping et al., in [2], has given an ontology-based approach for association rule mining. The existing traditional approach is not able solve the problem of useless rule mining and excessive concreteness of rules. In order to solve above problems better, association rule mining based-on ontology is used with the traditional apriori algorithm. Experimental results proved that ontology-based approach improved the efficiency of apriori algorithm.

Wang and Chen, in [3], have given an ontology-based approach for association rule mining. The apriori algorithm is the best known association rule mining algorithm whose objective is to find all co-occurrence relationships between data items. Performance of apriori algorithm degrades with the size of data. To overcome this problem ontology is used to represent the domain knowledge which reveals relationships between concepts. With the domain knowledge, the search space and counting time is reduced, so knowledge discovery is improved effectively and meaningfulness of hierarchical rules is increased.

2.2 Classification

Probowo et al., in [4], has given an ontology-based approach for enhancing automatic classification of web pages. Various challenges and issues on existing ontology-based approach are discussed. As the number of web data increasing daily, to classifying the entire web data manually without help of automated aid is not possible. Hence to help users to retrieve information relevant to their need ontology is used as a domain knowledge repository. Experimental evaluation proved that use of ontology improves accuracy as compared to existing technique.

Sundaramoorthy et al., in [5], has given an ontology-based approach for classification of user history. Users browsing history is used to meet the user need by classifying user in particular category. Existing approach degrades the performance due to lack of semantic knowledge about the user query. Hence ontology is used to understand the user query semantically. Experimental results proved that personalization using such ontology and semantic produce effective results.

2.3 Clustering

Wen et al., in [6], has given ontology-based clustering for improving genomic information retrieval. Traditional information retrieval is system based on basic language model having problem with synonym and polysemy. Hence in proposed method, ontology is used as background knowledge. Experimental evaluation proved that, proposed method solved the problem of synonym as well as polysemy up to greater extent as 6 compared to traditional language based model.

Hotho et al., in [7], has given ontology driven approach for text clustering. In the proposed approach, ontology used to represent the background knowledge about the domain during the preprocessing of results. Feature selection and aggregation techniques used with ontology. Based on these representations multiple clustering results are computed using K-Means algorithm. Then results are distinguished among various clusters and also relationships are maintained between the results by using ontology as domain knowledge. Use of ontology improves clustering of results and also allows selection between the results.

2.4 Information Retrieval

Tao et al., [8], has given personalized ontology model for used for web information gathering. The existing traditional approach was unable to retrieve the information as per user need. Proposed ontology based model represents user background knowledge for personalized web information gathering. This model constructs user personalized ontologies by extracting world knowledge and discovering user background knowledge from user local instance repositories. The proposed ontology model is evaluated by comparing against with benchmark models in web information gathering. The experimental evaluation proved that ontology-based model is superior and promising as compared to other models.

Saggion et al., in [9], has given ontology driven approach for business intelligence. Business intelligence process requires background knowledge to gather, merge, and analyze meaningful information from source of data. As a massive amount of data available on the web, hence extracting valuable information from the web is very tedious task for analyst. Without prior knowledge about the domain it's difficult to extract. In order to remove this drawback, ontology driven extraction system is developed. Performance measures had shown that, use of ontology as a background knowledge repository given a promising result than existing approach.

2.5 Prediction

Revoredo et al., in [10], has given a probabilistic ontology-based approach for semantic link prediction in a network. Due to semantic complexity in traditional machine learning algorithm, there is an uncertainty in link prediction. Hence probabilistic ontology-based approach used to provide the information about the domain to help in link prediction. In such schemes, numerical graph-based features and ontology-based features are computed; then both features are given as an input into a machine learning algorithm where prediction is performed. Experimental results proved that ontology-based model outperforms than existing prediction technique.

Caragea et al., in [11], has given an ontology-based approach for potential friendship link prediction in Live Journal social network. Existing approaches used in prediction is not able to capture the semantic similarity of the data. Hence the performance of the machine learning algorithm degrades. To overcome this problem ontology used as a training dataset to help machine learning algorithm. The experimental evaluation showed that ontology-based approach improves the performance of machine learning classifier at the task of predicting links in the social network.

All the ontology-based approaches in data mining techniques have given superior performance as compared to non-ontological approaches. Use of ontology as a background knowledge repository solved the problems of semantic complexity, time complexity, lack of training dataset, and relevancy of results up to much greater extent. Ontology as a domain knowledge repository has given much promising results in meeting the users need. Use of ontology has given new ray of hope to the data mining problem by solving them.

3. PROPOSED SOLUTION

The proposed approach describes the solution to the lack of domain knowledge. An existing system consist drawback of lack of domain knowledge and proposed system overcomes this drawback by using ontology as a knowledge repository. A knowledge base information retrieval is a technique that is said to exist if the domain knowledge is taken into the consideration. Source of domain knowledge is provided using domain ontology. Domain ontology is best one for such task. Necessary information about that domain is stored in ontology. Ontology acts as a background knowledge repository to information retrieval system. In this way meaningful information is retrieved which is hidden in the database. Use of ontology as a domain knowledge in various data mining techniques found effective for improving the precision of information retrieval.

3.1 Proposed System Architecture

Figure 2 shows architecture of the proposed system. Semantic query engine uses ontology as a source of domain knowledge. Ontology contains information about the domain in machine as well as human readable format. Proposed system mainly consists of THREE units - Semantic Query Engine, Transformation Unit and Query Processing Unit.

3.1.1 Semantic Query Engine

Semantic query engine is used to process the SPARQL queries. SPARQL is a data query language used to retrieve and manipulate data stored in ontology. Decision maker's requirements are directly represented in the form of SPARQL query. After executing the query, semantic query engine generates the output with the help of ontology. Generated output consists of background knowledge about the domain. Equation 3.1 shows process of semantic query engine.

$$f_x\,(Q_{sparql},\,O) \Rightarrow B_{knowledge} \qquad (3.1)$$

where, Q_{sparql}*: SPARQL Query*

O: Domain Ontology

$B_{knowledge}$*: Background Knowledge*

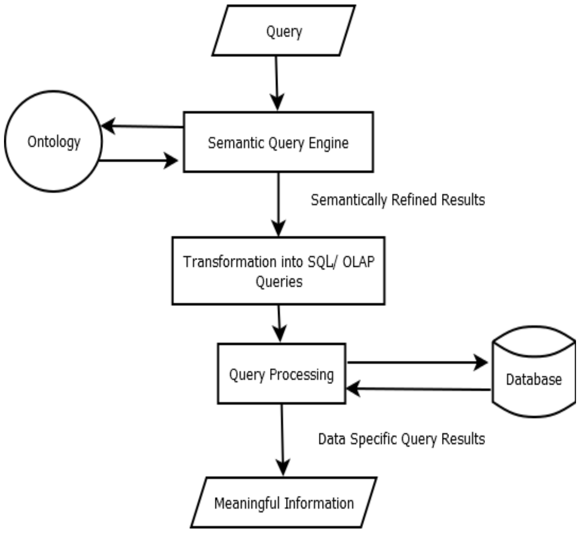
Query
Ontology
Semantic Query Engine
Semantically Refined Results
Transformation into SQL/ OLAP Queries
Query Processing
Database
Data Specific Query Results
Meaningful Information

Fig 2: Architecture of Proposed System

3.1.2 Transformation Unit

Transformation function constructs SQL query using background knowledge. Semantically enhanced results generated by semantic query engine are transformed into SQL or OLAP queries by using transformation function. Equation 3.2 shows working of transformation unit.

$$f_y\,(B_{knowledge}) \Rightarrow Q_{sql} \qquad (3.2)$$

where, $B_{knowledge}$ *: Background Knowledge*

Q_{sql}*: SQL Query*

3.1.3 Query Processing Unit

Query processing unit is used to process SQL queries generated by the transformation unit. Queries generated by transformation function are then executed on the database which contains historical or financial data about the organization. Results obtained by executing query on database are called as knowledge base information, which is then used for decision making. Equation 3.3 shows working of query processing unit.

$$f_z\,(Q_{sql},D) \Rightarrow M \qquad (3.3)$$

where, Q_{sql}*: SQL Query*

D: Database

M: Meaningful Information

3.2 Design

This section presents design of the proposed system. The design includes proposed solution for effective decision making by using ontology as a knowledge base. The design of proposed system is divided into TWO steps i.e. retrieval of background knowledge from the ontology and retrieval of knowledge base information by using ontology.

3.2.1 Retrieval of Background Knowledge from Ontology

Ontology is a collection of concepts, relations between the concepts, hierarchies between the parent class and subclass, properties of the concepts and axioms. Framework of ontology is represented in Equation 1.1. To retrieve background knowledge from the domain ontology, ontology is needed to be imported into the workspace.

import [Domain{C, R, H_c *, P, A}]*

where, C: Set of Concepts

R: Set of Relations

H_c*: Concept Hierarchy*

P: Properties or Attributes

A: Axioms

SPARQL query is used to retrieve the background knowledge from the ontology, as represented in Equation 3.1. Syntax of SPARQL query is very much similar to SQL query. SPARQL query uses select clause to select the triple and where condition to filter out the results based on the condition given by decision maker as input. Representation of the SPARQL query is as follows.

Prefix avg: <http//www.smanticweb.org/avg/avg/ontologies/2016/13/untitled − ontology – 12>

Select ?x where

{ ?x avg: WorksIn avg: Computer }

Prefix keyword is used to access attributes or data value attached to the uniform resource identifier (URI) from the ontology. Value of URI of respective ontology is stored in the *prefix* keyword. Select clause is used to select respective attribute from the triples. Keyword *where* is used to filter out the information based on the decision makers need.

After importing ontology SPARQL query is executed on the ontology by using semantic query engine. Semantic query engine used to retrieve the information from ontology.

$$f_{SQE}\,(Q_{sparql},\,Domain\{C, R, Hc , P, A\}) \Rightarrow RDF\ Node \qquad (3.4)$$

where, Q_{sparql}*: SPARQL Query*

Domain{C, R, H_c *, P, A}: Ontology*

RDF Node: Result of Semantic Query Engine

In Equation 3.4, Q_{sparql} is a SPARQL query, which is executed on the *Domain{C, R, Hc , P, A}* i.e. domain ontology by using f_{SQE} i.e. semantic query engine. Semantic query engine gives result in the form of *RDF Node* based on satisfying the where condition. *RDF Node* obtained contains URI attached with the literal or data value; which is used to formulate SQL query as shown in Equation 3.4. Example of retrieved result i.e. *RDF Node* is as follows.

http://www.semanticweb.org/avg/ontologies/2016/3/untitled-ontology – 12#Programmer

Part of the RDF Node before "#" symbol i.e. *"http://www.semanticweb.org/avg/ontologies/2016/3/untitled −ontology−12"* is called as URI and keyword *"Programmer"* is called as a literal or data value. Unnecessary part i.e. URI from the RDF Node is removed and literal is stored into another variable for further use as shown in Equation 3.5. RDF Node is removed to use literal as background knowledge with SQL query for a meaningful information retrieval.

$$\textit{getLiteral (RDF Node)} \Rightarrow \textit{Literal} \qquad (3.5)$$

where, RDF Node: Result of Semantic Query Engine

Literal: Keyword Extracted from RDF Node

The literal is collection of background knowledge about the domain. Extracted literal from Equation 3.5 is used for retrieving the knowledge base information from the raw data.

The Algorithm 1 shows sequence of flow of execution used in a proposed system for the retrieval of background knowledge from the ontology.

Algorithm 1

Retrieval of Background Knowledge from Ontology

```
Require: Ontology Domain(C, R, Hc , P, A),

Where C: Set of Concepts, R: Set of Relations, $H_c$: Hierarchies, P: Properties, A: Axioms.

Output: Literal \\ Literal is extracted from Ontology and used as background knowledge.

Let, String Prefix="Prefix avg:"; \\ This declares an string variable named Prefix used to store prefix URI of ontology.

String URI="<http://www.smanticweb.org/avg/avg/ontologies/2016/13/Domain.rdf>"; \\ This declares an string variable named Path used to store URI of ontology.

String Select="Select * where {"; \\ This declares a string variable named Select used to store select clause of the SPARQL query.

String Condition; \\ This declares a string variable named Condition used to store decision makers need.

String End="}"; \\ This declares a string variable named End used to terminate the query.

String Qsparql; \\ This declares a string variable named Qsparql to store the SPARQL query.

String Literal; \\ This declares an string variable named Literal to store the extracted value from RDF Node.

1: Start

2: import [Domain{C, R, $H_c$, P, A}] \\ Ontology is imported for querying into workspace.

3: input (Condition) \\ Input is taken from decision maker as per his need. e.g.  ?x avg: WorksIn avg: Computer

4: $Q_{sparql}$ ⇒ Prefix + Path + Select + Condition + End \\ $Q_{sparql}$ query is formed by concatenation.

5: $f_{SQE}$ (Sparql, Domain{C, R, Hc , P, A}) ⇒ RDFNode \\ Sparql query is fired on domain ontology by using semantic query engine.

6: getLiteral (RDFNode) ⇒ Literal \\ Literal value is extracted from RDF Node.

7: End
```

3.2.2 Retrieval of Knowledge Base Information using Ontology

Knowledge base information is retrieved from raw data by using background knowledge obtained from ontology. Respective table is loaded into the workspace, to retrieve knowledge base information from the table.

import [Domain{A, R, T}]

where, A: Set of Attributes

R: Set of Relations

T: Set of Tuples

To retrieve the knowledge base information from raw data, SQL query is used. SQL query is formulated using literal obtained from Equation 3.5. Literals are used as a filter to retrieve meaningful and relevant information from the raw data as per decision makers need. Literals are used as conditional keyword after where keyword.

$$Q_{SQL}\textit{=Select } * \textit{ From Table Where "Condition=Value (+Literal)";} \qquad (3.6)$$

where, Qsql: SQL Query

Literal: Extracted result from RDF Node

After formulating SQL query as shown in Equation 3.6, SQL query is executed on a table by using SQL query engine as shown in Equation 3.3. Results obtained after executing is knowledge base information used for decision making or problem solving.

$$f_{SQL}\,(Q_{sql}, \textit{Domain } \{A, R, T\}) \Rightarrow KBI \qquad (3.7)$$

where, Qsql: SQL Query

Domain {A, R, T}: Table containing raw data

KBI: Knowledge Base Information

As shown in Equation 3.7, f_{SQL} is SQL query engine used to execute SQL query i.e. Q_{sql} on table *Domain {A, R, T}*. After executing Q_{sql}, knowledge base information is retrieved as an output.

Abstract procedure of knowledge base information retrieval using ontology is given in Algorithm 2. The Algorithm 2 shows sequence of flow of execution used in a proposed system for retrieval of knowledge base information by using ontology.

Algorithm 2

Retrieval of Knowledge Base Information using Ontology

Require: Table Domain (A, R, T),

Where A: Set of Attributes, R: Set of Relations, T: Set of Tuples.

String Literal; \\ Value of Literal imported from Algorithm 1.

Output: Knowledge base information \\ Knowledge base information retrieved from table as a output and used further for decision making.

Let, String Select="Select $*$ From {"; \\ This declares an string variable named Select used to store select clause of the SQL query.

String where="where"; \\ This declares an string variable named where used to store *where* keyword.

String Qsql; \\ This declares an string variable named SQL used to store *SQL query*.

1: Start

2: import [Domain {(A, R, T)}] \\ Table is imported for querying into workspace.

3: import (Literal) \\ Literal from Algorithm 1 is imported.

4: Qsql $\Rightarrow$ Select + Domain + Where + Literal \\ SQL query is formed by concatenation.

5: f_{SQL} (Q_{sql}, Domain {A, R, T}) $\Rightarrow$ Knowledge Base Information.
\\SPARQL query is fired on domain ontology by using semantic query engine.

6: End

4. RESULT AND DISCUSSION

The proposed system is evaluated via experiments on the real time example i.e. on student domain. The real time example domain is chosen in such a way that prediction and decision making is used to solve the problems related that domain. Results obtained from the experiments are then used for the evaluation of proposed system. Detailed information about example case is as follows.

Student domain is chosen to make a prediction about the future of the student such as marks in the upcoming exams, carrier related prediction etc. Now days, number of students are widely growing and playing key role in the success of every organization. So for every academic organization to keep track of each student is necessary, in order to predict the future of student and maintain the quality of organization. Decisions are made using student related information such as percentage obtained in SSC, HSC, 1st to 3rd year of graduation, family income, distance he/she is commuting from home to college etc. The parameters selected on the basis of criteria, which affects directly and indirectly on the student's academic performance. By analyzing student's previous marks, academic records and demographic information, upcoming exams result is predicted. Ontology is used to provide the background information about the student domain. Various parameters which are affecting the student's performance are stored in the ontology. RDF graph representation of the student domain ontology is as shown in Figure 3.

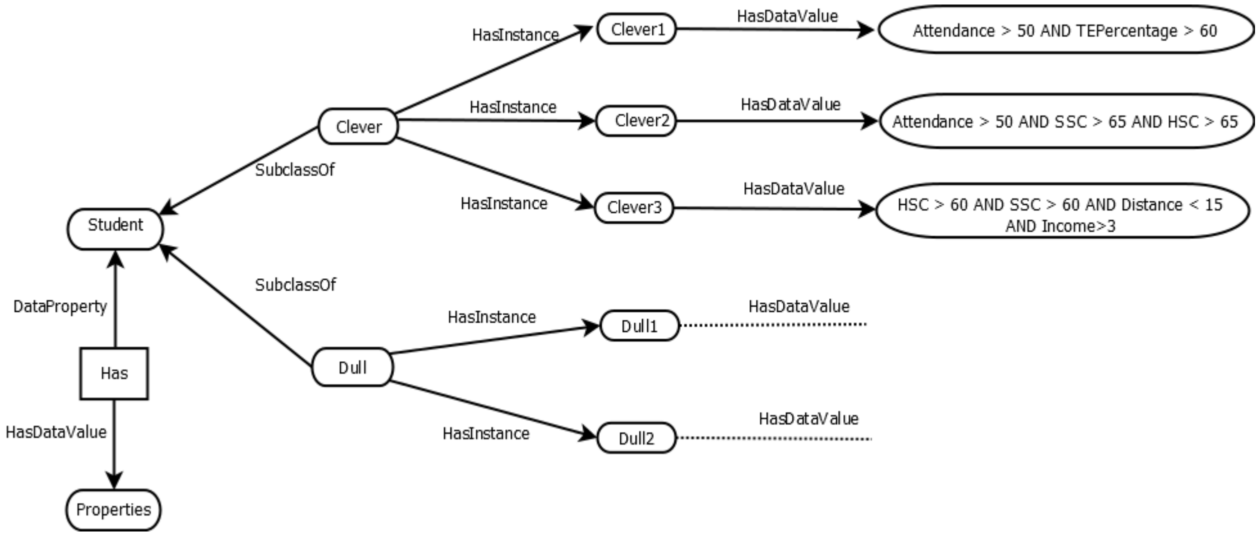

Fig 3: RDF Graph Representation of the Student Domain Ontology

To predict the result, students are categorized into TWO subclasses i.e. Clever and Dull students. Various conditions are applied to sort them by using their information. Clever students are categorized into *"Attendance > 50 AND TE Percentage > 60"* i.e. students who have attendance greater than 50% and percentage greater than 60 in 3rd year of graduation as Clever1 instance, *"Attendance > 50 AND SSC*

> 65 AND HSC > 65" i.e. students who have attendance greater than 50% and percentage greater than 65 in SSC and HSC as Clever2 instance and *"HSC > 60 AND SSC > 60 AND Distance < 15 AND Income > 3"* i.e. students who have percentage greater than 60 in SSC and HSC, distance commuting is less than 15km and family income is greater than 3 lacks per annum as Clever3 instance. Similarly, students are categorized into Dull category.

Experimental evaluation is carried out on the basis of student information system consist of 1000 records sample each. Let us consider the database about the student information system of particular organization. Student information system is categorized into THREE tables named *Personal Information Table, Academic Information Table and Examination Result Table*. *Personal Information Table* contains general information about the student such as student id, name of the student, gender, date of birth, branch, SSC percentage, HSC percentage, distance he/she is commuting and annual family income. *Academic Information Table* contains performance related information about the student such as student id, name, unit test 1 and unit test 2 marks and attendance of the student. Examination *Result Table* contains annual exam performance related information about the student such as student id, name and annual percentage of first year to third year of graduation.

4.1 Result

From the student information system as mentioned above, find out that how many students pass in their final year exams irrespective toughness of the question paper. To make a decision, only necessary information attribute such as student id, SSC and HSC percentage, attendance, percentage of third year graduation, distance and income are selected and merged into a single table called as *Decision Table* for further use.

4.1.1 Results Obtained using Existing Approach

Existing approach uses SQL query to retrieve the meaningful information from the table. Background knowledge about the domain is not taken into the consideration by the existing approach. SQL query is executed on the *Decision Table*.

SQL Query ⇒
SELECT * From Decision Where Attendance > 50 AND TEPercentage > 60;

Result ⇒

Number of students will pass in final exam irrespective of the toughness of the question paper = 584.

4.1.2 Results Obtained using Proposed Approach

Proposed approach uses SPARQL query to retrieve background knowledge from the Student Domain Ontology as shown in Figure 3. SPARQL Query 1, SPARQL Query 2 and SPARQL Query 3 are used to retrieve the background knowledge of Clever1, Clever2 and Clever3 instances respectively as shown below.

SPARQL Query 1 ⇒
Prefix avg: <http://www.smanticweb.org/avg/avg/ontologies/2016/13/untitled-ontology-21#>

Select ?x

where {avg:Clever1 avg:Has ?x};

Result ⇒

Attendance > 50 AND TEPercentage > 60

SPARQL Query 2⇒

Prefix avg: <http://www.smanticweb.org/avg/avg/ontologies/2016/13/untitled-ontology-21#>

Select ?x

where {avg:Clever2 avg:Has ?x};

Result ⇒

Attendance > 50 AND SSC > 65 AND HSC > 65

SPARQL Query 3 ⇒

Prefix avg: <http://www.smanticweb.org/avg/avg/ontologies/2016/13/untitled-ontology-21#>

Select ?x

where {avg:Clever3 avg:Has ?x};

Result ⇒

HSC > 60 AND SSC > 60 AND Distance < 15 AND Income > 3

Results obtained from *Student Domain Ontology* are then used to retrieve the meaningful information from the Decision Table. Students qualified in the Clever1 instance; stored in respective table and assigned with 1 as a confidence. Students qualified in the Clever2 instance; stored in respective table and assigned with 0.6 as a confidence. Students qualified in the Clever3 instance; stored in respective table and assigned with 0.4 as a confidence. Confidence is a weightage value i.e. estimate calculated with a probability to tell true chances that respective student pass his exam irrespective of toughness of the question paper. Then union of students qualified in Clever1, Clever2, Clever3 tables is performed with the duplicates are allowed and stored in the respective table. Then confidence value is summed up to each other on the basis of same ID. To find out who will pass in the final exam irrespective of toughness of the paper; threshold is applied to the confidence value and stored in the respective table. Students who have confidence value greater than 1; pass in the final exam irrespective of the toughness of paper.

Result ⇒

Number of students will pass in final exam irrespective of the toughness of the question paper = 534.

4.2 Discussion

In student domain example, query is to find out how many students pass in the final exam irrespective of the toughness of question paper. In existing approach, information retrieved from the database is based on the condition *"Attendance > 50 AND TE Percentage > 60"*. Students who have attendance greater than 50% and percentage greater than 60 in his third year of graduation are selected by a existing approach. Out of 1000 students, 584 students are selected by using existing approach. In proposed approach, Student Domain Ontology is used to provide the background knowledge about the student domain. Information retrieved using proposed approach is based on the THREE conditions i.e. *"Attendance > 50 AND TE Percentage > 60", "Attendance > 50 AND SSC > 65 AND HSC > 65" and "HSC > 60 AND SSC > 60 AND Distance < 15 AND Income > 3"*. Out of 1000 students, 534 students are selected by using proposed approach.

Result analysis shows that results obtained using proposed approach is more meaningful and relevant as compared to existing approach. Results of proposed approach are based on the various important parameters such as percentage in HSC and SSC, family income, distance commuting daily to come college; which affects the performance of student. These parameters are used as background knowledge in the form of ontology. However, these parameters are also possible to store in the database and corresponding SQL query will provide the information. But the information retrieved from the database by using SQL query does not include the experience. However, using ontology the experience is added with the information which defined as knowledge. Hence, use of ontology as a source of background knowledge is better as compared to database. In general also, ontology is superior as compared to database as a source of background knowledge.

Database is generally used to store the data of specific application or organization. Ontology is specially designed and invented to provide knowledge, semantics and concepts about the domain, hence it's easier to store required background knowledge in the ontology. Database focuses on the data only not on the knowledge, but focus of ontology is on the understanding the concepts, relations and knowledge about the domain. Ontology provides the knowledge in machine as well as human understandable format, so it's easy to for a human as well as machine to work with the ontology. Maintenance of the ontology is easier as compared to the database. Hence ontology is used as source of background knowledge.

5. CONCLUSION

Knowledge base information retrieval is the process of retrieving meaningful information from the database. In traditional approach, meaningful information remains hidden in the database itself due to lack of domain knowledge. To make the decision making more effective, ontology is used as a domain knowledge in the proposed system. Experimental evaluation carried out on the results obtained by using both ontological and non-ontological approach. Analysis of results shows that, information obtained by using proposed approach is more meaningful and relevant as compared to the existing approach. The proposed system shows that, involvement of ontology gives promising and superior results than the existing approach. The proposed system mostly focuses on the decision making as well as the future prediction by using past or historical data.

In the future work, the focus will be on improving the efficiency of recommender system by using ontology as a source of semantic knowledge related to the concepts.

6. REFERENCES

[1] T. R. Gruber, "Toward principles for the design of ontologies used for knowledge sharing," International Journal Human Computer Studies, vol. 43, no. 5-6, pp. 907–928, 1995.

[2] W. Xuping, N. Zijian, and C. Haiyan, "Research on association rules mining based-on ontology in ecommerce," Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing, vol. 2, no. 31, pp. 3542–3549, 2007.

[3] Y. Wang and Y. Chen, "A new association rules mining method based on ontology theory," Proceedings of the Conference on Advanced Computational Intelligence, vol. 2, no. 6, pp. 287–291, October 2012.

[4] R. Prabowo, M. Jackson, P. Burden, and H.-D. Knoell, "Ontology-based automatic classification for the web pages: Design, implementation and evaluation," Proceedings of the 3rd International Conference on Web Information Systems Engineering, pp. 182–191, 2002.

[5] P. Sundaramoorthy, M. Sreekrishna, S. Bhuvaneshwari, and M. Selvam, "Ontology based classification of user history in obscured web search," Proceedings of the 2nd International Conference on Current Trends in Engineering and Technology, pp. 258–261, July 2014.

[6] J. Wen, Z. Li, and X. Hu, "Ontology based clustering for improving genomic ir," Proceedings of the Twentieth IEEE International Symposium on Computer-Based Medical Systems, pp. 225–230, 2007.

[7] A. Hotho, S. Staab, and A. Maedche, "Ontology-based text document clustering," Springer Science and Business Media, vol. 16, no. 4, pp. 48–54, 2012.

[8] X. Tao, Y. Li, and N. Zhong, "A personalized ontology model for web information gathering," IEEE Transactions on Knowledge and Data Engineering, vol. 23, no. 4, pp. 496–511, April 2011.

[9] D. M. Horacio Saggion, Adam Funk and K. Bontcheva, "Ontology-based information extraction for business intelligence," Springer Berlin Heidelberg, vol. 4825, pp. 843–856, 2007.

[10] K. Revoredo, J. E. O. Luna, , and F. G. Cozman, "Semantic link prediction through probabilistic description logics," Journal of the Brazilian Computer Society, pp. 397–409, 2013.

[11] D. Caragea, V. Bahirwani, W. Aljandal, and W. H. Hsu, "Ontology based link prediction in the live journal social network," Proceedings of the Eighth Symposium on Abstraction, Reformulation, and Approximation, pp. 34–41, 2009.