A Rule-Based Natural Language Interface to Data Warehouse

Saliha Zahoor[1] and Fiaz Majeed[2]

[1]Department of Computer Science, University of Gujrat (UOG), Gujrat, Pakistan
[2]Department of Information Technology, University of Gujrat (UOG), Gujrat, Pakistan
[1]saliha.zahoor@uog.edu.pk, [2]fiaz.majeed@uog.edu.pk

***Abstract*— Data Warehouse Systems are used for decision-making. It is difficult task for casual users to write their request in technical language because they may not have knowledge of technical structure of Database or Data Warehouse. Writing questions in user's natural languages are easy. In this work, a Natural Language Interface to Data Warehouse has been presented. A set of rules are going to be proposed to understand accurate aggregations in user input natural language question. With the help of rules, the aggregation elements are precisely identified from the user query. Finally, empirical analysis has been carried out to evaluate the proposed system. With the help of this mechanism, system can easily find the aggregate elements from the user input query.**

***Keywords*— Natural Language Interface, Aggregate functions, Data Dictionary, Data Warehouse and Mapping Rules**

I. INTRODUCTION

The main objective of this research is the identification of accurate aggregation elements from the user input query which is written in natural language format. When user input query in natural language format (English language is scope of this work), system split it into keywords. It performs splitting for the purpose of matching. According to best of our knowledge, a very little research is carried out to find accurate aggregate elements from the user input natural language query.

The application rules are created for the matching of user input query keywords. The keywords are matched by the given list of the mapping rules. By application of these rules, it becomes easier to identify the aggregation elements from the query. The aggregation elements include dimensions, facts, dimension attributes, fact measures, aggregation functions and grouping sequence.

Further, a dictionary is maintained that hold synonyms of domain keywords. The keywords of query are initially searched in the dictionary. Later these are matched with the application rules. After successful matching of keywords with mapping rules, aggregation elements are returned. Finally, identified elements are mapped into Online Analytical Processing (OLAP) query which is executed and results are retrieved.

This paper is arranged in this order: section II gives literature survey which provides information about the natural language processing, existing natural language interfaces and their comparison. Later, Section III describes architecture of the system and each of its components in detailed form. Section IV discusses mapping rules while Section V presents system evaluation and research findings. Finally, section VI concludes the work and provides future directions.

II. LITERATURE SURVEY

By Natural Language Processing (NLP), useful information is accessed from user input question and results are generated accordingly [1]. NLP provides easy and attractive way of communication between the users and the computer. With the use of NLP, the users can easily generate their questions and can retrieve required information. Natural Language Interfaces is a broad term now and is of immense importance due to its simplicity and modern simple ways for human being to convert their inputs towards the system in their own language. It has been the area of research to help people interacting with various systems and to facilitate the technology. The objective of Natural Language Interface is to access desired results of the users in their native language. Natural Language Interfaces to Databases (NLIDB) is an area in which user natural language query is converted into the query well and truly understandable by the system. Work on such study has been in progress well back in sixties but yet this is still an attractive area to be explored. This will benefit us in many ways like it can help save time as well as better commands and quick work. So this is still an area to work on [2].

The system LUNAR [3] was brought in 1971 and built for the purpose of seeking answers of questions regarding rock samples. These samples were collected from the moon. This system uses two databases for two different jobs i.e. Substance analysis and text reference. This was quite an achievement for this system. According to [4], the LUNAR system managed to respond 78 % queries without an error and further improved the bar to 90% when dictionary mistakes were eliminated. Woods procedures and Augmented Transition Network parser have been used in the LUNAR System. By using questions, Hendrix introduced in 1978 the databases for changing NLI into the information recognized by the databases. But the down side of LIFER/LADDER was that the system was only able to understand with restricted join conditions [5]. In 1977, Philips Question Answering system used various layers to rectify syntactic parser errors.

This system is composed of a syntactic parser having three split layers in it. The three layers are Database Language, World Model Language and English Formal Language [6]. Database administrator with the help of TEAM was effortlessly configuring the database with no information of NLIDBs [7], [8]. A big division of investigation of that point in time was committed to portability issues. The improvement of NLIDBs was carried on and several such systems have been developed. These systems include Banks [9], Discover [10], DBXplorer [11], and others [12], [13].

In a large data warehouse, the single most impressive way to influence presentation is an appropriate record of aggregate functions that jointly exist with the main records. Aggregates directly influence in major cases speeding queries of one hundred or even up to one thousand [14]. Aggregate navigator functionality should be implemented to explore the right table with the right grain when requests to Data Warehouse are made. The numbers of possible aggregations are determined by every possible combination of dimension granularities. By monitoring queries, we can decide that which aggregations will match our query patterns [15].

Rules are regulation or principles that are made to do work under some criteria. Rules are used to represent the knowledge or information in some allowable condition. In data mining field, different association rules are used. These association rules are Multilevel, Multidimensional and Quantitative. The knowledge of different level of abstraction is represented in the multilevel association rule, more than one dimension involvement is represented in the multidimensional rule and the numeric values of the attributes are represented in the Quantitative association rules. There is certain classification of rules in data mining. Further, rules are built from the Decision trees, case-based reasoning, and lexical analyzing. There are different ways of developing the rules on the basis of condition and requirements [16].

III. SYSTEM ARCHITECTURE

User input query is parsed down into keywords. With the support of synonyms maintained in the Data Dictionary, keywords are matched to elements by using the application rules. User input query is finally converted into Structured Query Language (SQL) query. The system executes the query and generates result. The proposed system architecture is shown in Fig. 1.

The detail of each component is given below:

A. Natural Language User Input Query

The user input the query in natural language form (such as English) users have no knowledge to write the queries in the SQL syntax. The query input interface of this system is similar to Google search engine.

B. Query Parser

The Query Parser breaks the natural language query into keyword on the basis of space character. So the query is split down into keywords and a list is generated.

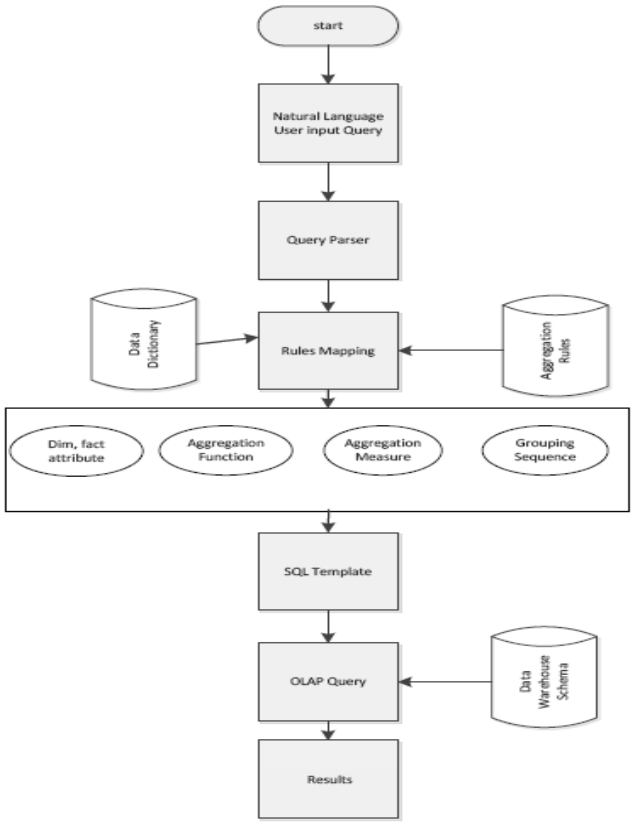

Fig. 1. Architecture of the proposed system

C. Data Dictionary

Data dictionary is a collection of words generated for the elements exist in logical schema. It contains set of synonyms for each schema element. It further includes synonyms of aggregation functions. The user query keywords are searched in the Data Dictionary to retrieve target schema element. We have developed a dictionary for the elements that are present in the Data Warehouse schema. In addition, Data Dictionary is updated at run-time as unknown keywords are appeared in the query. The synonyms of such unknown keywords are generated from Word Net.

D. Aggregation Rules

The aggregation rules are stored in three component format i.e., Rule, Rule notation and Rule explanation. The detail of aggregation rules is given in section IV.

E. Rules Mapping

The Mapping rules are made to map the user query statement. We have made rules for the recognition of the tables and attributes words in the user input statement. When there is a Dimension/Fact table word in the query that specific table is mapped with the help of rules. With the help of mapping rules, the aggregate words are also identified (AVG, MIN, MAX, COUNT, SUM and so on) in the user query and synonyms of the aggregate functions. The OLAP query is finally built according to identified elements with the use of rules.

The procedure of rules mapping is as follows: Each keyword and its synonyms maintained in the Data Dictionary are matched with the rules. The result of rules mapping returns aggregation elements i.e., aggregation functions, Dimensions/Fact Tables, attributes and measures.

F. Aggregation Elements

1) Dimension/Fact Attribute

The dimensions and fact tables, their attributes and measures respectively are identified with the help of dictionary and application rules.

2) Aggregation Function

To identify the aggregate function from the user input query, the application rules have been generated. These application Rules automatically identify the aggregate functions. Here Aggregation Functions like AVG (), MIN (), MAX (), COUNT (), SUM () are recognized.

3) Aggregation Measure

The synonyms are generated for fact measures and application rules are also stored to precisely identify them.

4) Grouping Sequence

The correct grouping sequence is generated to format results according to the requirement of the user.

G. SQL Template

We have developed a general SQL template that has the following syntax for the template.

Select attributes, aggregation functions/aggregation Measures
From dim/fact tables
Where conditional expressions
Group by Grouping Attributes

H. Data Warehouse Schema

The Data Warehouse logical schema is used to match the query keywords with schema elements. System directly accesses the schema during keyword matching and explores it with the help of Domain Dictionary and mapping rules.

I. OLAP Query

After identification of aggregation elements, system maps them in the SQL template. In next step, OLAP query is easily built which is executed on OLAP engine.

J. Results

The generated OLAP query using SQL template is executed and results are displayed.

IV. STRUCTURE OF MAPPING RULES

The mapping rules have been developed for the schema. The dimension and fact tables that are available in the schema, rules have been developed for the tables and the attributes of tables. With the help of these rules, the tables and attributes of the tables are identified from the user input query. In this way, required tables and attributes of the tables can accurately be identified. Mapping rules for one of the dimension is given Table I.

The mapping rules have been developed for the Aggregate functions and synonyms of the aggregate functions. Some rules of aggregate function Max ( ) and its synonyms are depicted in Table II.

V. EXPERIMENTS AND RESULTS

The proposed method has been evaluated on the AdventureWorksDW which is data warehouse developed in MS SQL Server. For experiments, core i3 machine has been used with 2 GB RAM. The part of Data Warehouse schema is depicted in Fig. 2.

TABLE I. MAPPING RULES FOR THE PRODUCT DIMENSION TABLE

Rule	Rule Notation	Rule Explanation
EnglishProduct Name	attrib-EnglishProduct Name	Rule to show attribute EnglishProduct Name of Product Table
English Product Name	attrib-EnglishProduct Name	Rule to show attribute EnglishProduct Name of Product Table
english product name	attrib-EnglishProduct Name	Rule to show attribute EnglishProduct Name of Product Table
englishproduct name	attrib-EnglishProduct Name	Rule to show attribute EnglishProduct Name of Product Table
SpanishProduc tName	attrib-SpanishProduc tName	Rule to show attribute SpanishProduc tName of Product Table
FrenchProduct Name	attrib-FrenchProduct Name	Rule to show attribute FrenchProduct Name of Product Table

TABLE II. MAPPING RULES FOR THE MAX () FUNCTION

Rule	Rule Notation	Rule Explation
greater	af-MAX()	Rule show aggregate function max()
highest	af-MAX()	Rule show aggregate function max()
Higher	af-MAX()	Rule show aggregate function max()
most	af-MAX()	Rule show aggregate function max()
biggest	af-MAX()	Rule show aggregate function max()
maximum	af-MAX()	Rule show aggregate function max()

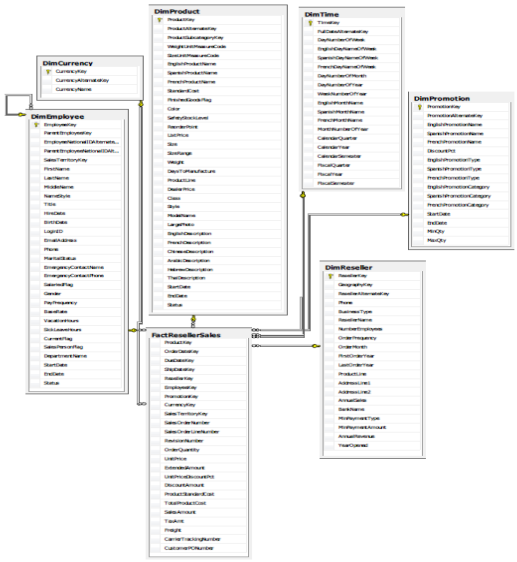

Fig. 2. Data Warehouse schema

The Analysis of Aggregation functions and Synonyms of the Aggregation function is performed in Table III. The five aggregation functions and its synonyms are taken. Thus most relevant synonyms are taken that are required and suitable for the Aggregate functions. The results are also shown graphically in Fig. 3.

TABLE III. ANALYSIS OF SYNONYMS FOR AGGREGATION FUNCTIONS

Aggregation Word	Total Synonyms	Selected Synonyms
SUM()	31	11
AVG()	12	10
MAX()	32	17
MIN()	16	12
COUNT()	17	6

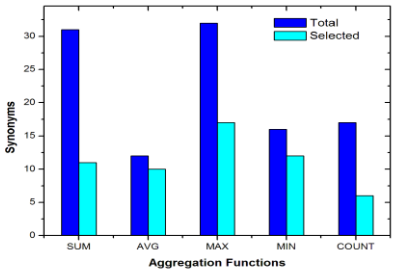

Fig. 3. Analysis of Query Mapping Process

VI. CONCLUSIONS

In this work, a Natural Language Interface to Data Warehouse has been proposed. The architecture of the system provides detail about participating modules in the system. It develops rules based on underlying schema with the support of the domain dictionary. The rules are then used to interpret natural language query. Empirical analysis has been performed to evaluate the proposed system. As future work, it is required to generate limited rules according to user interestingness measure.

REFERENCES

[1] Warschauer M., Healey D., "Computers and language learning: An overview", Language Teaching, 31:57-71, 1998.

[2] Androutsopoulos G.D. Ritchie, Thanisch P., "Natural Language Interfaces to Databases – An Introduction", Journal of Natural Language Engineering 1 Part 1, 29–81, 1995.

[3] Woods W., Kaplan R., Webber B., "The Lunar Sciences Natural Language Information System", Bolt Beranek and Newman Inc., Cambridge, Massachusetts Final Report. B. B. N. Report No 2378.1972

[4] Woods W. "An experimental parsing system for transition network grammars", In Natural language Processing, R. Rustin, Ed., Algorithmic Press, New York, 1973.

[5] Hendrix G., Sacrdoti E., Sagalowicz D., Slocum J., "Developing a natural language interface to complex data", ACM Transactions on Database Systems, 3(2): 105 – 147, 1978.

[6] Scha R.J.H., "Philips Question Answering System PHILIQA1", In SIGART Newsletter, no.61. ACM, New York, 1977.

[7] Grosz B.J., "TEAM: A Transportable Natural-Language Interface System", In Proceedings of the 1st Conference on Applied Natural Language Processing, Santa Monica, California, 39–45, 1983.

[8] Grosz B.J., Appelt D.E., Martin P.A., Pereira F.C.N., "TEAM: An Experiment in the Design of Transportable Natural-Language Interfaces", Artificial Intelligence, 32: 173–24, 1987.

[9] Bhalotia G., Hulgeri A., Nakhe C., Chakrabarti S., Sudarshan S., "Keyword Searching and Browsing in Databases using BANKS", In ICDE, 431–440, 2002.
[10] Hristidis V., Papakonstantinou Y., "DISCOVER: Keyword Search in Relational Databases", In VLDB, 670–681, 2002.
[11] Agrawal S., Chaudhuri S., Das G., "DBXplorer: A System for Keyword-Based Search over Relational Databases", In ICDE, 5–16, 2002.
[12] Hristidis V., Gravano L., Papakonstantinou Y., "Efficient IR-style keyword search over relational databases, In *VLDB*,850–861, 2003.
[13] He H., Wang H., Yang J., Yu P.S., "BLINKS: Ranked Keyword Searches on Graphs", SIGMOD'07, June 11-14, Beijing, China, 2007.
[14] Aggregate Navigation With (Almost) No Metadata". [08-15, 1995] http://www.kimballgroup.com/1996/08/02/aggregate-navigation-with-almost-no-metadata/, 1995.
[15] Ralph Kimball et al., "The Data Warehouse Toolkit", Second Edition, Wiley Publishing, Inc., ISBN 978-0-470-14977-5, Page 355, 2008.
[16] Jiawei Han and Micheline Kamber.,"Data Mining Concepts and Techniques" second Edition by Elsevier Inc 2006.

Saliha Zahoor is MS scholar and functioning as Lecturer in University of Gujrat- Gujrat Pakistan since September 2007. During her 6 years stint as lecturer at university of Gujarat she is been involved in helping her students on various research issues. She is taking research projects on NLI for data warehouse and on software engineering as well. This paper is the result of her constant efforts on NLI for data warehouse project and is a part of the uncontaminated work done on her final thesis report. She started working on this paper in early 2012.

Fiaz Majeed received MS degree from COMSATS Institute of Information Technology (CIIT) Lahore Pakistan in 2009. He is currently PhD scholar in University of Engineering and Technology (UET) Lahore Pakistan. Further, he is Lecturer in University of Gujrat (UOG), Gujrat, Pakistan and working on couple of research projects. His research interests include data warehousing, data mining, data streams and information retrieval. He has published more than 10 research papers in refereed journals and international conference proceedings in the above areas. This paper is part of the research project on the development of Rule based NLI for Data Waerhouse and he is supervisor of this project. It was initiated in spring 2012 at University of Gujrat.