A Rule Based Graphical User Interface to Relational Database using NLP

Abhijeet R. Sontakke[1], **Amit Pimpalkar**[2]

[1]Department of Computer Science and Technology, G.H. Raisoni Academy of Engineering and Technology, Nagpur, India

[2]Professor, Department of Computer Science and Technology, G.H. Raisoni Academy of Engineering and Technology, Nagpur, India

Abstract: ***Database management systems have been used for storing and retrieving the data. Databases are very hard to use since their interfaces is rigid in cooperating with users. Almost all e-governance applications are using databases. They are providing services like weather forecasting, agriculture, banking, railway etc. people who are snug with Hindi language needs these applications to accept Hindi sentence as a query, process it and after execution provide the result to the user in Hindi language itself. Natural language processing (NLP) plays an important role while working on Hindi language to make it more efficient for the common people. Here we have developed the rule based system which will satisfy the user need and it will accept Hindi language as query and gives out put in Hindi language only.***

Keywords: DBMS, HLIDBMS, NLP, NLIDB, SQL

1. Introduction

NLP is a branch of artificial intelligence which includes Information Machine Translation, retrieval, Language Analysis. Accessing database by natural language processor is to make dataset access easier for the common people. While natural language may be the easiest or hardest as per the different category of the people. It is the system to learn and use, it has proved to be the hardest to a computer to master. A user must have the knowledge of Structured Query Language (SQL) and DBMS to access the database. It is very much limited for only those users who have the knowledge of SQL and other database related languages and they can access data or information very easily. So in order to access the information, a rule based graphical user interface is used which requires some basic training for using this system. In India, many people know English but are not that much fluent enough to formulate queries in it. With the help of this rule based graphical user interface an end user can query the system in natural languages such as English, Hindi and Marathi etc., and can see the result in the same language.

We have developed a system for people who know Hindi language. User can access database using Hindi language and get the result in the same language. For example सभी विद्यार्थी का नाम, अंक बताओ. For our system we are going to use relational database as a test case.

2. Related Work

In the present computing world, computer based information technologies have been extensively used to help many organizations, private companies, academic and education institutions to manage their processes and information systems. Information systems are used to manage data. A information management system that is capable of managing all the kinds of data, stored in the database is known as Database Management System (DBMS) [1]. Databases are comprehensive element in private and public information systems which are essential in number of application areas [2]. Databases are very much important for large variety of application areas employing private and public information systems. Retrieval of a large amount of the same type of data is very efficient in relational databases [3], but still the user has to master the DB schema completely to formulate the queries.

(SQL) Structured Query Language is an ANSI standard for manipulating and accessing the information stored in databases. It is comprehensively employed in industry and is supported by major DBMS. Most of the languages used for manipulating relational database systems are based on the norms of SQL. In the past few years many advances have been made in the field of databases and many other fields of critical relevance to information technology. An intelligent database is an emerging database technology that has dramatic impact on the way we think and work [4].

In recent times, there has been a rising demand for non-expert users to query relational databases in a more natural language encompassing linguistic variables and terms, instead of operating on the values of the attributes. Talking to a computer in a natural language such as plain English is always a dream that drives the progress of human-computer interaction work [5, 6].

For the last thirty years, numerous attempts have been made to build useful natural language interface. It has turned out to be much more difficult than what was originally expected. There have been large numbers of research works introducing the theories and implementations of NLIDBs. There are mainly four kinds of four NLIDBs framework [7]. The first type of framework is based on pattern matching. A typical application of this type of framework is SAVVY [8]. In this system, various patterns are written in some different kind of queries and these patterns are executed after the complete queries are entered. The main advantage of pattern matching approach is that no elaborate parsing and modules of interpretation are required and the systems are very easy to implement. Some pattern matching systems were able to perform impressively well in certain applications. One of the NLP system that is based on pattern-matching approach is

ELIZA [9] However as mentioned [7], a pattern matching system is too shallow and therefore would often lead to bad failures. LUNAR [10] system supports a syntax-based approach where a parsing algorithm technique is used to regenerate a parse tree depending on user's different type of queries. This method is specially used in application-specific relational database systems. A query language must be provided by the system to start the mapping from parse tree to the database query. However, it is very difficult to decide the mapping rules from the parse tree for some intermediate representation languages can be used to convert the statements in natural language to a well known formal query language. MASQUE/SQL [11] is a very good example for this approach. It is a front-end language for relational databases that can be reached through SQL.User defines the types of the domain for which database refers using a hierarchy in a built-in domain-editor. However, words expected to be appear in queries with their logical predicates are also declared by the user itself. Queries are first transformed into a Prolog-like language LQL, then into structured query language (SQL). The system generating the logic queries is independent from the database and therefore, it is very flexible in domain replacements is the advantage of the system.

Despite of the achievements attained in this area, present day NLIDBs do not guarantee correct translation of queries in natural language to database languages. The most desirable characteristic of the NLIDBs that the researchers are proposing is Domain Independence, which means that interface can be used with different databases and reconfiguration of the NLIDB from one domain to another is done automatically[12,13].

B. Sujata etal [14] introduced a concept that, the SQL norms are been pursued in almost all languages for relational database management systems. However, not everybody is able to write SQL queries as they may not be aware of the structure of the database. So this has led to the development of Natural Language interface for databases. There is an overwhelming need for non-sophisticated users to query relational databases in their natural language instead of working with the syntax of SQL. As a result many natural language interfaces to databases have been developed, which provides different options for manipulating queries. The idea of using Natural Language instead of SQL has prompted the development of new type of processing called NLIDB. Mohil Dua etal [15] had implemented the system based on NLP which gives output on the basis of NLIDB and Hindi language interface to database management system that give the proper result for only select, update and delete queries. Ashish kumar [16] has given the system which is based on Hindi language interface to database using semantic matching.

3. Proposed System

3.1 Problem Statement

Hindi language interface to relational database is completely based on the rules through which we are going to perform the operations like select, insert, update, delete. We are also working to provide the advance query operation such as functionality of aggregate functions such as MIN (), MAX (), SUM () and AVG (). The user will type the query in Hindi language and that natural language has been processed and will give the output in Hindi language only. Time difference has been calculated, system will give translation time and execution time in milliseconds as well as in nanoseconds.

3.2 System Overview

This system has been developed for the people who are not having any knowledge of query languages. Asking query in Hindi language and getting the result in same language is the biggest advantage for the person who knows Hindi language. Objectives that are covered are given as:-

Design the GUI through which user can easily give input query in Hindi language also he will get output in the same language. We have design the Hindi language interface to relational database. We have developed a system that can handle Hindi query for the extraction of single or multiple columns from tables stored in databases. This system can perform all the operations such as select, insert, update, delete on relational databases as well as it will perform aggregate functions such as sum (), max (), min (), avg () on the same relational database. We have also worked on semantic behavior of the independent query. Translation time and execution time is also calculated whenever the query fired.

3.3 Methodology

To achieve the above objective methodology used is given as- we are going to use the rule based system which will follow and execute each and every query as per the rules made for it. First it will identify the nature of the query i.e. select, update, delete, create, insert and also it will identify that the query is with aggregation functions or not. We are using the relational database so it is very much flexible we can easily store all Hindi as well as English values in it and also we can easily retrieve it. Randomize automatic record generation technique is also there so that we can easily generate maximum number of records in very less time. Appropriate mapping of tokens with database values should be done by extracting table, columns information from input Hindi sentences. With the help of stored values of databases generate SQL query by mapping input query. Finally we will execute the Hindi query and also get the output in Hindi language itself.

3.4 Implementation & Architecture of the system

Architecture of Hindi language interface to relational database using NLP is given and explained below from fig1. This architecture is known as HLIDBMS i.e. Hindi Language Interface to Database management System. There are important phases i.e. Tokenizer, query type rule, query table rule, basic queries and its sub rules, query generator engine DBMS & database server. In tokenize phase Hindi sentence is split into tokens. This is done with fact that all the tokens are separated by a space gap from each other. All the tokens which we get in this phase are stored in an array. Tokens are words of Hindi language. Token may be a table name, column name, condition, any value, command name, operation name or any non-useful word. To understand this; let the user query is as:

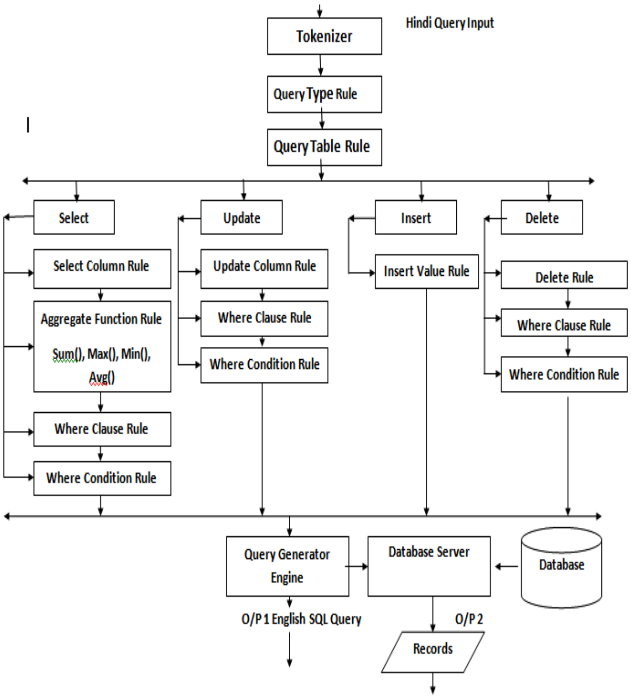


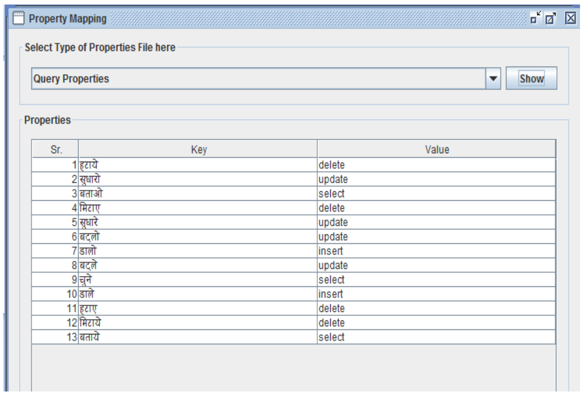
Figure 1: Architecture

सभी विद्यार्थी का नाम,अंक बताओ. This Hindi sentence has 7 tokens. First token is सभी which is the starting of sentence. Now सभी means it is reflecting like select all i.e. in SQL we say 'Select *', another token is विद्यार्थी it is reflecting the name of the database table i.e. 'student table' Some tokens may be fields name as in the above query नाम and अंक are the field names. There is conjunctions also like का as well as we also included the comas (,) in the list of tokens & finally last thing is बताओ which is reflecting as the 'select query' Therefore after this step we have all the tokens from which the sentence is composed of.

After that we will apply the query type rule. Query type rule is a rule which will identify which type of query it is whether it is select, insert, update, delete type of query. We are given with the query properties through which we can easily identify the associated Hindi word which is given in the sentence within a query and is given below in figure 2.

Later it will identify the table name with the help of query table rules. It will just see whether the given table is present there or not. These both the things have been possible because of the tokenizer and its tokens which we are matching under each rule. Once the query rules and table rules has been applied then we will proceed with the further tokens and we will apply the sub rules of the selected query.

If the query in Hindi will be the select query then it will look for the rules like column rules, aggregate function rule, where clause and where condition rule.



Select Type of Properties File here

Query Properties

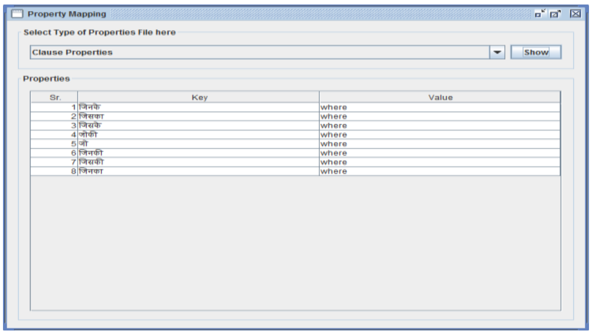
Properties

Sr.	Key	Value
1	हटाये	delete
2	सुधारो	update
3	बताओ	select
4	मिटाए	delete
5	सुधारे	update
6	बदलो	update
7	डालो	insert
8	बदले	update
9	चुने	select
10	डाले	insert
11	हटाए	delete
12	मिटाये	delete
13	बताये	select

Figure 2: Query Properties

It will work with the help of tokens only. It is like column rules it will select the number of columns given in the Hindi query.

Aggregate function will identify whether it is min (), max (), sum (), avg () query or not . Other rules like where clause for that we are given with the properties i.e. It will identify all the associated Hindi English words which are given below in the fig 3.



Select Type of Properties File here

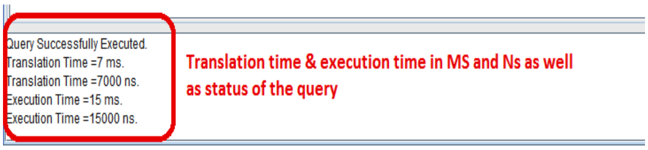
Clause Properties

Properties

Sr.	Key	Value
1	जिनके	where
2	जिसका	where
3	जिसके	where
4	जोकी	where
5	जो	where
6	जिनकी	where
7	जिसकी	where
8	जिनका	where

Figure 3: Where clause property

Similarly where condition is also there it will work like same as given above it is consisting of all the conditional part and its associated Hindi words including <,>,=,logical and ,or not etc. Similarly for update query it is having update column rule ,where clause rule and where condition rule and its working is same as explained above. The same way insert and delete also work. At last there is query generator which will generate query from Hindi sentence .that query generated will be fired to database and all the selected records selected rows has been displayed in Hindi Language. SQL is generated in this phase according to Hindi sentence. Execute query and display result to user the above SQL query is executed and result of which in Hindi language is displayed to user. The output is in the form of Hindi language and we are giving query also in Hindi language and processing of all this has been done by inner module as explained above.



```
Query Successfully Executed.
Translation Time =7 ms.
Translation Time =7000 ns.
Execution Time =15 ms.
Execution Time =15000 ns.
```

Translation time & execution time in MS and Ns as well as status of the query

Figure 4: GUI & timing results

Once the query has been executed and the result has been shown side by side it will also show the timing result which include whether the query has been successfully executed or not if it is failed it will show the unsuccessful message as shown in the fig4. It will also give the translation time in milliseconds as well as nanoseconds to notice the minute difference during conversion and same in the case of execution time also, it will show the time required to execute the query.

4. Conclusion

Rule based graphical user interface to relational database is presented in this paper. The system will accepts Hindi sentence as a query and gives output in Hindi itself. It is very much useful for the people who do not have any prior knowledge of database and SQL queries languages. We are using different rule along with the NLP to perform operation such as insert, update, delete, select as well as the aggregate functions such as min (), max (), sum (), avg () etc. This system can be enhanced by making it more generic. We can also implement it for very complex queries like join operations & order by operations (queries). To make the system more friendly the dialogue based system can be used in which user will provide the input Hindi query through speech interface.

References

[1] Zongmin Ma, "Intelligent Databases: Technologies and Applications", IGI publishing, 320 pages, 2007.

[2] Burlesan, Donald K., Joe Celko, John Paul Cook, and Peter Gulutzan. 2003. Advanced SQL Database Programmer Handbook. BMC Software and DBAZine.

[3] Dietmar Wolfram, "Applications of SQL for Informetric Data Processing", Proceedings of the 33rd conference of the Canadian Association for Information Science, 2005.

[4] "IntelligentDatabase"from Http://searchsqlserver.techtarget.com/sDefinition /0,,sid87_gci1124415,00.html

[5] M. Brodie, “Future Intelligent Information Systems: AI and Database Technologies Working Together” in Readings in Artificial Intelligence and Databases, Morgan Kaufman, San Mateo, CA, 1988.

[6] Donald P. Mckay and Timothy W. Finin, "The Intelligent Database Interface: Integrating AI and Database systems", In Proceedings of the 1990 National Conference on Artificial Intelligence, pp. 677--684, 1990.

[7] Androutsopoulos, G. Ritchie, P. Thanisch, Natural language interfaces to databases – an introduction, Journal of Natural Language Engineering 1 (1) (1995) 29–81.

[8] Johnson T. (1985), Natural Language Computing: The commercial Applications. Ovum Limited, London.

[9] Weizenbaum J, (1966).“ELIZA- A computer program for the study of natural language communication between man and machine”,Communications of the ACM, 9(1), pp.36-45.

[10] W.A. Woods, R.M. Kaplan, B.N. Webber, The lunar sciences natural language information system: Final report, BBN eport 2378, Bolt Beranek and Newman Inc., Cambridge, Massachussetts,1972.

[11] Androutsopoulos, G. Ritchie, P. Thanisch, MASQUE/SQL – An efficient and portable natural language query interface forrelational databases, in: Proceedings of the Sixth International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, Edinburgh, 1993, pp. 327–330.

[12] Rangel, Rodolfo and Joaquín Pérez, O. and Juan Javier González, B. and Gelbukh, Alexander and Sidorov, Grigori and Rodríguez, M.(2005), “A Domain Independent Natural Language Interface to Databases Capable of Processing Complex Queries”,MICAI 2005: Advances in Artificial Intelligence LNCS 3789 pp 833-842.

[13] B. Juan J. González, Rodolfo A. Pazos Rangel, I. Cristina Cruz C.,H. Héctor J. Fraire and L. de Santos Aguilar, et al.(2006) "Issues in Translating from Natural Language to SQL in a Domain-Independent Natural Language Interface to Databases"LNCS 4293,MICAI: Advances in Artificial Intelligence, pp 922-931.

[14] B. Sujata, S. Viswanadha Raju and Humera Shaziya, “A Servey of Natural Language Interface to Database Management System” International Journal of Science and Advance Technology”, vol.2, no. 6,june 2012 .

[15] Mohit dua, Sandeep Kumar, Zorawar Singh Virak, “Hindi Language Graphical User interface to Database Management System” 12th International Conference of Machine Learning and Application 2013.

[16] Ashish kumar , “Hindi Language Interface to Database using Semantic Matching” Oriental journal of Computer Science and Technology vol 6 , no(2) pgs 133-140 Issue 0974-6471 ,june 2013.

[17] Abhijeet R. Sontakke, Prof. Amit Pimpalkar “A Review Paper on Hindi language Graphical User Interface to Relational Database using NLP” International journal of Advanced Research in Computer Engineering & technology (IJARCET)volume 3 Issue 10, October 2014

References

Abhijeet Ramrao Sontakke, pursuing M-tech from G.H. Raisoni academy of engineering and technology Nagpur, Maharashtra, India. Field of interest is NLP (Natural Language Processing) and Object Oriented Systems.

Prof. Amit Pimpalkar, Assistant Professor at G.H. Raisoni Academy of Engineering & Technology, Nagpur. Area of interest is Natural Language Processing, Data Mining, DBMS and Image Processing.