International Journal of Scientific Research in Computer Science and Engineering
Vol.6, Issue.1, pp.50-55, Feb (2018)

Research Paper

E-ISSN: 2320-7639

Keyword based Marathi Interface to the Database using Natural Language Processing

A.B. Patil[1*], S. C. Pawar[2]

[1*]Dept. of Information Technology , Rajarambapu Institute of Technology, Rajaramnagar, India
[2]Dept. of Computer Engineering, K.J.Somiaya College of Engineering, Mumbai, India

[*]*Corresponding Author: ashwini.patil@ritindia.edu, Tel.: +919970700994*

Available online at: www.isroset.org

Received 30/Dec/2017, Revised 15/Jan/2018, Accepted 29/Jan/2018, Online 28/Feb/2018

***Abstract*—** Now a day's a computer plays a vital role in almost all the sectors/application. The available information is stored and retrieved in/from the database. Database Management system (such as SQL, Oracle) allows to handle the database by creation of the database, querying the database, updating database, and administration of databases. So there is need of skilled person (database experts) in order to deal with database management system, but this is not always true. The common people with lack of expertise are there to handle the database, who doesn't know about the syntax/format of queries to be fire on the database. Also they are more comfortable in their native languages like "Marathi", "Hindi", "Arabic". It is essential to have the database interface in native language (e.g Marathi) so that the non-expert person can also interact with database for storing and extracting the data. In this paper we are proposing the Marathi language interface using Natural Language Processing (NLP) to handle the database. This interface can handle the queries in native language which may be syntactically and semantically incorrect and convert it into correct form.(e.g SQL query).

Rest of the paper is organized as follows, Section I contains the introduction of natural language processing , Section II contain the related work of natural language interface to the database system, Section III contain the problem definition and proposed system, Section IV contain the architecture and algorithm of proposed system, section V explain the working of proposed system with example, Section VI give conclusion with future direction.

***Keywords*—** Natural Language Processing, SQL, Keyword based interface, Natural language interface

I. INTRODUCTION

In this computer era, most of sectors like education, banking, government and medical become computerized. Information is stored in the database. Most all applications store and extract the data from database by using the Structured Query Language (SQL) language. The person who has knowledge of this SQL language can able to access the database. But people working in above mentioned sectors are might not be the expert in SQL language. So it's important to use query in natural language to interact with the database. But queries in natural form not understand by the database. So need to focus on NLP concept to convert queries (in Natural language) into SQL. Using natural language interface person with no knowledge of SQL can also handle the database easily.

One of the most important applications of Natural Language Processing (NLP) is Natural Language Interface to the database (NLIDB). This interface allows the non-database expert to interact with the database in natural language.

Natural Language Processing:
Natural language processing is the branch of Artificial Intelligence (AI), which concern with computer can process the human like languages. Science goal of NLP is to understand how language produced and how it is understand. Also engineering goal of NLP is to reduce the man- machine interaction.

Stages of NLP:
Following are the stages of NLP:-

- *Phonetics and Phonology:* Study of words and related sounds.
- *Morphology*: Word formation rules form root word.
- *Lexical Analysis:* Refers to dictionary access and obtain properties of the word.

- *Syntax analysis :*Assign the grammatical correct structure to the sentence
- *Semantic Analysis:* Representation of knowledge.
- *Pragmatics:* Modeling user intension.
- *Discourse planning:* Processing of sequence of sentence.

Applications of Natural Language Processing:
There are various applications of NLP which includes,

- Text-to-Speech
- Automatic speech recognition
- Information Retrieval
- Information Extraction
- Question answering
- Spelling error correction
- Automatic Summarization
- Text Categorization
- Text Mining
- Part of Speech Tagging.
- Natural language Interface to DB

II. LITERATURE SURVEY

LUNAR:
W. Woods, et al [1] designed system named as LUNAR in 1971. LUNAR answer questions about the Apollo 11 moon rocks for the NASA Manned Spacecraft Center. As it handles the 90% of queries without fail, the performance of LUNAR is excellent. The LUNAR system uses two databases for chemical analysis and literature references. Also it uses an Augmented Transition Network (ATN) parser and Woods' procedural Semantics [2].

LADDER:
The LADDER [3] has been developed by Earl D. Sacerdoti for the management help to Navy decision makers. The LADDER consists of three major components. First is INLAND (for Informal Natural Language Access to Navy Data) which access the user queries in natural form and produce the database queries without any knowledge about how data is stored in various files. So second component called IDA (Intelligent Data Access) is used to break down the query against the data base into a sequence of queries against various files and also compose those queries in the language of the DBMS. Data-language produced by IDA does not refer to specific files in specific directory on specific machine. It produce generic DBMS queries which is given to the third component called FAM (for File Access Manager) which is used to show where the files are located throughout the distributed database. It then establishes connections to the appropriate computers, opens the files and transmits the Data-language query for execution. This system can handle the queries against a data base consisting of some 14 files containing about 100 fields.

CHAT 80:
David H.D, et al [4] has designed system for natural language question-answer called as Chat-80. Chat-80 translates English questions into the Prolog subset of logic. CHAT -80 contains basic facts about 150 countries of the world, oceans, major seas, major rivers, major cities and vocabulary of English words for querying the database. The questions within a limited subset of English can be translated into a certain subset of logic which, when suitably transformed, is executable as efficient Prolog code.

The translation from English sentence to logical form involves slot filling, scope determination and parsing. One limitation is, while converting English sentence into logic, 'pronouns' are not taken into account. After converting sentence into logical form, it is possible to answer the question due to precise semantics of logic.

Basically, Chat-80 do the augmentation of logical form of a query with extra control information, to make it efficient piece of Prolog program, which can be executed to answer the query.

NLKBIDB:
Axita Shah , et al [5] proposed new system of natural language and keyword based interface to agricultural database, which gives result for syntactically correct or incorrect queries. NLKBIDB is combined approach of NLIDB and KBIDB. NLIDB is more accurate than KBIDB for generation of SQL query based on natural language, but it can't handle syntactically incorrect queries which are handled by KBIDB. System mainly consists of three agents i.e Natural Language Agent , Keyword Based Agent and Knowledge Base Agent along with SQL generator and SQL executor. Natural Language Agent perform lexical , syntactic and semantic analysis of query entered by user in natural form and generated tree is given to SQL generator. Keyword Based Agent do mapping of token generated from lexical analysis with knowledge. Knowledge Base Agent generated the knowledge base using representing the metadata in XML form. The SQL generator work according to rules of natural language or rules of keyword based interface.

Using Natural Language Processing in Order to Create SQL Queries:
F.Siasar djahantighi, M.Norouzifard, S.H.Davarpanah, M.H.Shenassa, et al [6] proposed new system work with the model natural language interface using database and designed to framework owl ontology. The author expresses the brief idea of relational database with

ontology and use of word net dictionary. The paper gives the characteristics of frameworks.

- Natural language and database interface is organized in such way that business model and conceptual model of database works with ontology.
- The framework uses lexicons with the synonymy and antonym relations to integrate with word net. The framework maps the lexicons entered by the user with the domain lexicons.
- This framework uses the Discourse Representation Structure (DRS), and Discourse representation Theory (DRT).

This structure uses template techniques to convert DRS to structured query language. To accomplish this, the format is determined by the SQL grammar and content is provided by DRS. With respect to above defined features the framework is divided into two modules. One module is language processing module and second module is database processing module. Language processing module is getting completed with different sub module as, semantic grammar, syntax parser, lexicon, semantic interpreter and most important is Ontology. Again this ontology is work with respect business model and database schema. Language processing module accepts input as natural language as input and it into tree representation and converts it into intermediate representation as DRS. Natural language query sentences are translated into DRS query sentences, and then translated into SQL query sentences.

Hindi Language GUI to DBMS:

Mohit Dua, et al [7] implemented interface to database system in Hindi language. Architecture of the system mainly consists of three component i.e tokenizer, mapper and query generator. Tokenize separate the token from the user query which is in Hindi. Then mapper maps these tokens to the English tokens available in the lexicon along with type i.e command, function, condition_start, column_name, table_name etc.Query generator generate the SQL query according the mapping done. This system was support the various queries that include joining operator, logical operator, selection of multiple column etc.

Enhancing the Relevance of Information Retrieval by Querying the Database in Natural form:

Prof. Amisha Shingala, et al [8] has developed Student Interface System in Natural Language (English). The system can convert the user question in English form into SQL query. The English sentence is first converted into intermediate logic by semantic interpreter and then converted into SQL form to retrieve data from database.

III. PROBLEM DEFINITION

In most of the sectors today, there is use of application based software's. These applications can store and retrieve information to and from database. For most of application relational database management system is used. A special-purpose programming language designed for managing data held in a relational database management system (RDBMS) i.e SQL (structured query language). Anybody who interacts with database should be expert in SQL, so as to store and retrieve, update and delete the records from the database. That person, who knows the syntax of all queries in SQL, can handle the database properly. But everywhere it's not possible. The person who operates the database is not always expert in SQL. Another important issue is, people are more comfortable in their local natural languages rather than database language i.e SQL. So it creates the problem while handling the database. And because of that they can't use the application software properly.

Interface to database system:

One of the solutions to above mentioned problem is design and develops a software interface which can handle the queries in local natural language, which can be syntactically and semantically incorrect. By using this software interface, it is possible to convert the natural language query statements written in local language into required form i.e SQL, which can understand by the database. Because of this interface it is not necessary to have expert person to handle the database.

Two types of interface are allow the user to interact with the database i.e natural language interface to DB(NLIDB) and Keyword based interface to DB(KBIDB).

I) Natural language interface to DB(NLIDB):-

It consist of three main components that language component, intermediate representation language and database component.

a) Language Component:

Language component consist of,

Lexical Analyzer:

Query entered by user in natural form is given as input to the lexical analyzer. The Lexical analyzer is act as string tokenizer, which will generate tokens from the given query by considering space as delimiter excluding the stop words.

Syntax Analyzer:

Syntax analyzer, parse the given input according to the grammar and assign the correct parse tree to input query (sentence)

Semantic Analyzer:

Semantic analyzer, represent the meaning of linguistic input in formal structure like First Order Predicate Calculus (FOPC). This process is called as meaning representation. The meaning of the sentence is depends on the meaning of the parts of sentences i.e word. The meaning of a sentence is not based solely on the words that make it up, it is based on the ordering, grouping, and relations among the words in the sentence.

Discourse Analysis:

Processing of sequence of sentences is known as discourse and it requires discourse planning.

Pragmatics:

Study of how the context provides meaning is known as pragmatics. It's main aim to modeling user intention.

b) intermediate representation language:

It's an interface between the front end language component and database component. It is kind of knowledge base which is derived from the metadata of the database.

c) Database component:

Database component uses this intermediate language to convert the natural language query into SQL query.

II) Keyword based interface to DB (KBIDB):-

Keyword based interface to database involves two important steps

i) Generation of symbol table (keyword table) in database which shows the equivalence between the word in natural language to word in English word. Basically it's creation of lexicon, which stores the tokens in natural language and correspond English token along with types to indicate whether it is table name, column name, value…etc.

ii) Preparation of SQL query from natural language query by mapping the keyword from natural query to keywords stored in the symbol table.

IV. PROPOSED WORK

In this paper we are focusing on Marathi language. Our proposed system can accepts the queries in Marathi language and convert that query into SQL format, which is executed against the database and finally result will be displayed to user. The system will handle SELECT, UPDATE and DELETE type of queries on the database.

The proposed work consist of four phases namely ***String tokenizer , Keyword Mapper, Query Generator and SQL Query Executor.***

V. ARCHITECTURE OF PROPOSED WORK

The following figure shows the architecture of the proposed work.

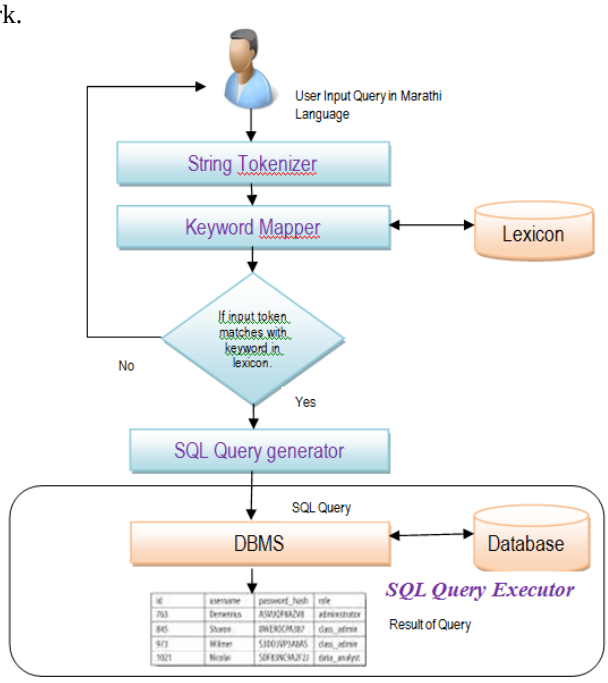

Figure1: Architecture of Proposed System

Proposed system consist different module like string tokenizer, Keyword Mapper, Query Generator and SQL Query Executor.

User query in the natural form is given input to the **string tokenizer**. The String tokernizer split the sentence into different token. Words separated by the white space are known as tokens. These token are stored in one array. Token of sentence may indicate type of command, table name, column name, value, start_of_condition, condition, logical_operator which are the important parts of SQL query.

The token which was extracted from the sentence is then compared with lexicon dictionary which contain token (word) in Marathi language, it's corresponding English word and type of the token indicating either that token is table_name, column_name, value, condition, start_of_condition, logical_operator or command of operation performed on the database.

The **Keyword mapper** compare the Marathi tokens with lexicon directory to extract the corresponding keyword of English language along with type. All this information given to the SQL query generator.

SQL query generator is main module of the proposed system, which is responsible for actual query generation. Based on knowledge of SQL query and information given by the

keyword mapper, SQL query is generated which is further executed by SQL query executor against the database to display the result to the end user.

VI. ALGORITHM OF PROPOSED WORK

```
Start
Input query in natural language from User.
len=Number of words in the given query
Split the given sentence into tokens and store it into array.
for(i=0 to len)
        If word of index i with type "command" found in lexicon then
            Command= English token corresponds to the word
        If word with type start_of_condition found in lexicon then
            Condition= English token corresponds to the word
            x=Index of that word in an array
loop
  for (i=0 to x-1)    //Selection Part
        {
                Map each word in array with lexicon directory.
                If word of index i with type table_name found in lexicon then
                        table_name=English token corresponds to the word
                For each word of index i with type column_name found in lexicon
                        cn=1
                        column_name(cn)=English token corresponds to the word
                        cn++
        }

   for(i=x+1 to len)    //Condition Part
        {
                if word of index i found in lexicon with type "column_name"
                        cn=1
                        condition_column_name(cn)=English token corresponds to the word
                        cn++
                else if word of index i found in lexicon with type "condition_part"
                        con_col=1
                        condition_column_name(con)=English token corresponds to the word
                                                    after condition_part
                        con_col++
                else if word of index i found in lexicon with type "condition"
                        con=1
                        condition (con)=English token corresponds to the word like >,<, >=,<=
                                    and =.
                        con++
                else if word of index i found in lexicon with type "logical_operator"
                        logocal_operator=English token corresponds to the word
                else
                        v=1
                        value[v]=English token corresponds to the word
                        v++
        }
End
```

VII. WORKING OF PROPOSED SYSTEM

Our proposed system can accept the queries in Marathi language and convert that query into SQL format. The system will handle SELECT, UPDATE and DELETE type of queries on the database.

For testing purpose we have taken following assumption,

i) Student Database is taken into consideration with following attributes,
   Name, Roll_No, Class, City, Mobile_No, Marks

ii) It's assume that we have the lexicon dictionary which contain the mapping of Marathi word to corresponding English token along with the types of token.

The proposed system will work as follows,

a) Simple Select query for single column selection

First we consider a very simple query entered by user in Marathi language as follows,

Query Entered by User: ***sava- ivaVaqyaaMcaI naavao saaMga.***

This query is given input to the string tokenizer ,which will separate the token from the query. In entered query total 4 token are there, generated token is stored in the array, which will given to Keyword mapper. The mapper search in the array for the command. Here in this example the word "***saaMga"*** is the command which will map to "select" word in English. Also identified column_name1 is "Name" by mapping "***naavao*** " to the English token and the Marathi word "***ivaVaqyaaMcaI"*** will map to student.

The identified mapped words will given input to the query generator. Based on the knowledge base and output from string tokenizer mapper will generate the following SQL query,

Corresponding SQL Query: ***"Select Name from Student"***

b) Select query for multicolumn selection

We will consider the another example where user has entered query for selection of multiple column.

Query Entered by User: **sava- ivaVaqyaaMcaI naavao , kmaaMk , vaga-- saaMga.**

Total 6 token are identified in above entered query. In the given Marathi query mapper identify the different column names like Name, Roll_No and Class after mapping Marathi word to English token which will be stored in column_name1, column_name2 and column_name3, corresponds to Marathi word naavao , kmaaMk , vaga--.

Corresponding SQL Query: **"Select Name, Roll_No, Class from *Student"***

c) Select with where clause

Let's consider example with where clause

Query Entered by User: sava- ivaVaqyaaMcaI naavao saaMga jyaaMcaa vaga-- tRtIya vaYa- Aahooo

In the query entered by the user total 9 token are there. The mapper identify the column name as "naavao" and store into variable column_name1 as name. Also it find lexicon with type "condition" as "jyaaMcaa" which will map to "where" clause. The mapper store column_names after condition into

condition_column_name and remaining part into value array. It will also consider the logocal_operators.

Corresponding SQL Query: **"Select Name from Student where Class = *TY"***

d) Select with function

Query Entered by User: sava- ivaVaqyaaMcaI naavao saaMga jyaaMcaa gaUnaaMcaI sarasarI caaLIsa pooxaa jyaast Aaho.

The given query works similar like query with where clause. The word "sarasarI" is mapped to function AVG , "caaLIsa" will map to 40 and stored into condition_column_name and "jyaast" will mapped to ">" and stored into condition_array.

Corresponding SQL Query: **"Select Name from Student where AVG(Marks)>40*"***

VIII. CONCLUSION

Keyword based Marathi interface to Database system (KBMIDB) is a system that accepts queries in Marathi language. User give Marathi query to the database interface which will converted into SQL query.

When user enters the query in Marathi, the sentence is parsed and all the tokens are stored in an array. Then mapping of these tokens will be done to find out table name, column name, conditions, commands and values. Then these tokens are converted into English word. From these words, SQL query is formulated. The system supports all type of selection, updating and deletion on database.

REFERENCES

[1] W. Woods, R. Kaplan,"The lunar sciences natural language information system: final report" Published in 1978.

[2] B.Sujatha, Dr.S.Viswanadha Raju and H. Shaziya , "A Survey of Natural Language Interface to Database Management System" International Journal of Science and Advanced Technology (ISSN 2221-8386) Volume 2 No 6 June 2012.

[3] E. D. Sacerdoti , "Language Access to Distributed Data with Error Recovery"

[4] H. D. David Warren and Fernando C. N. Pereira, "An Efficient Easily Adaptable System for Interpreting Natural Language Queries" American Journal of Computational Linguistics, volume 8, Number 3-4, July-December 1982

[5] A.Shah, Dr. J. Pareek, H. Patel and N. Panchal , "NLKBIDB - Natural Language and Keyword Based Interface to Database" , 978-1-4673-6217-7/13 ,2013 IEEE

[6] F.Siasar, M.Norouzifard1, S.H.Davarpanah2, M.H. "Using Natural Language Processing in Order to Create SQL Queries" Proceedings of the International Conference on Computer and Communication Engineering at Kuala Lumpur, Malaysia,

[7] M. Dua, S. Kumar, Z. Singh Virk, "Hindi Language Graphical User Interface to Database Management System" , 12th International conference on Machine Learning and Application, 2013.

[8] Prof. A. Shingala and Dr. P. Virparia, "Enhancing the Relevance of Information Retrieval by Querying the Database in Natural form" , International Conference on Intelligent Systems and Signal Processing (ISSP),2013

Authors Profile

Prof. Ashwini B. Patil is has complted MTech in Computer Scieice and Technolgy. She is currently working in Rajarambapu Institute of technology, Rajaramnagar,,Sangli (*Maharashtra*) . She has 13 yr. of techaing experience. Her area of research is Natural language processing ,Networking, Wireless Sensor Network, Internet of Things. She has published a book on "Commence Web Development with PHP and MySQL"

Prof. Swapnil C. Pawar has completed M.E. in computer engineering. He is currently working as Assistant professor in K.J. Somiaya college Engineering, Mumbai (Maharashtra). HE has 7 years of teaching experience. His area of interest in programming language and Web technology. Also He is working on App development and Machine learning area for his research.

