

# TELUGU LANGUAGE INTERFACE TO DATABASES

D.Ramesh<sup>1</sup>, Suresh Kumar Sanampudi<sup>2</sup>

Associate Professor, Department of IT, JNTUHCEJ, Karimnagar, INDIA<sup>1</sup>

Assistant Professor, Department of IT, JNTUHCEJ, Karimnagar, INDIA<sup>2</sup>

**Abstract:** The abundant information available on internet generates the need to store data in an organized manner so that searching, retrieving and maintaining of data becomes easier. Databases is a technology that stores the data in an logical and organized manner. To efficiently operate these databases, knowledge of structures query language (SQL) becomes essential. But the usage of SQL restricts the access to databases from the users who don't have the knowledge of them. A need for interface comes into the picture to enable the access of these databases even to a non-expert users. This paper describes the design to develop Telugu language interface to databases. Bank database is used as a case study to develop telugu language interface. The performance of the system has shown to be satisfactory.

**Keywords:** Natural Language Interfaces, Databases, Language Processing, Parsing

## I. INTRODUCTION

Language is the primary means of communication used by humans. It is the tool we use to express the greater part of our ideas and emotions. Learning new concepts and expressing ideas through them is so natural that we hardly realize how we process natural language. Natural Language Processing (NLP) is concerned with the development of Computational models of aspects of human language processing. There are two main reasons of such development:

- ✓ To develop automated tools for language processing.
- ✓ To gain a better understanding of human communication.

Building computational models with human language processing abilities requires Knowledge of how humans acquire store and process language. It also requires the knowledge of world and of language. Companies have related the problem of extracting data from a Data Base Management System (DBMS) by using the tools like MS Access, Oracle and others. A person with no knowledge of Structured Query Language (SQL) may find himself or herself handicapped in corresponding with these tools. Thus it creates the need to develop products for people to interact with the database in their own native language.

In Andhra Pradesh (AP) Telugu is used as a source of language for communication. We have developed a model

for Telugu language interface to enable the people of AP to use the databases in their own native language. In this paper we used Banking database as a case study to test the working of our model. The remaining sections of the paper are organized as follows: section 2 describes the related techniques followed earlier to develop the natural language interface. Section 3 elaborates the algorithm developed for telugu language interface. Section four provides the experimental results. Section 4 concludes the paper.

## II. RELATED WORK

Several NLP researchers have been working on the topic of Natural language Interface to Databases (NLIDB). Different techniques such as pattern matching, syntax based and semantic based system are used to develop NLIDBs[4][1]. Some of the early NLIDBs relied on pattern-matching techniques to answer the user's questions. The main advantage of the pattern-matching approach is its simplicity: no elaborate parsing and interpretation modules are needed, and the systems are easy to implement. Also, pattern-matching systems often manage to come up with some reasonable answer, even if the input is out of the range of sentences the patterns were designed to handle. One of the best natural language processing system that role in this style is SANVY[2].

In syntax-based systems the user's question is parsed (*i.e.* analyzed syntactically) and the resulting parse tree is directly mapped to an expression in some database query language. Syntax-based systems use a grammar that describes the



possible syntactic structures of the user’s questions. Syntax-based NLIDBs usually interface to application-specific database systems that provide database query languages carefully designed to facilitate the mapping from the parse tree to the database query. It is usually difficult to devise mapping rules that will transform directly the parse tree into some expression in a real-life database query language. LUNAR is a good example of syntax based systems [5]

In semantic grammar systems, the question-answering is still done by parsing the input and mapping the parse tree to a database query. The difference, in this case, is that the grammar’s categories do not necessarily correspond to syntactic concepts. Semantic information about the knowledge domain is hard-wired into the semantic grammar that’s why systems based on this approach are very difficult to port to other knowledge domains a new semantic grammar has to be written whenever the NLIDB is configured for a new knowledge domain. Semantic grammar are used in one of the NLIDB systems like LADDER[3].

**III. ALGORITHM**

Telugu Language Interface (TLI) to database system provides an interface to the user which helps him/her to query the database in Telugu. As the query languages like SQL are very difficult to use for the common people and they find it very hard to learn. So, to make database applications easy to use for these people who don’t know query languages, Telugu Language Interface for databases has been developed. In this paper baking database is used as case study to test the working of our model.

In this model a query is asked in Telugu language for retrieving the relevant information from database. To achieve this task an algorithm is developed which follows the step by step procedure to generate the SQL query from a given query in Telugu.

*Algorithm TLI*

*Step1: Input the query in Telugu Language*

*Sep 2: Tokenize the query*

*Step 3 If token==Lookup(table\_names)*

*else GOTO Step 7*

*Step 4 If token==Lookup(Column names)*

*else GOTO Step 7*

*Step 5 If token++Lookup(conditions)*

*else GOTO Step 7*

*Step 6 Generate SQL Query*

*Step 7 Invalid Input type of Query*

*Step 8Execute the query*

*Step 9 Represent the Result in Telugu*

User provides input in the form of simple sentence to the system in Telugu language. The queries of three different

types are allowed such as: Queries for selection of whole table, Queries for selection of certain columns. Queries for selection of certain rows from certain columns *i.e.* queries with “where” condition Tokenizer convert the sentence into into word level tokens. These tokens are mapped to identify the table name from the list of words stored in the look up table. If a match is found, then the column names are identified. The existence of conditional clause is also identified to generate the query of the formatSelect “Column names” from “table name” where “condition”.

**IV EXPERIMENTS AND RESULTS**

Experiments are conducted by taking the bank database as a case study. Three different models of queries are executed whose structures of query generations are shown in the table below.

TABLE 1  
Query for selection of all columns

| Natural Language Query          | SQL Query               |
|---------------------------------|-------------------------|
| వనియోగదారులందరి వివరాలు తెలపండి | SELECT * FROM customer; |

TABLE2  
Queries for selection of certain columns

| Natural Language Query   | SQL Query                         |
|--------------------------|-----------------------------------|
| వనియోగదారుల వీధి తెలపండి | SELECT cust_street FROM customer; |

TABLE 3  
Queries for Selection of certain rows and certain columns

| Natural Language Query                    | SQL Query  |
|---|--|
| కమలనగర్ యొక్క పట్టణం తెలపండి              | SELECT cust_city FROM customer<br>WHERE cust_street="కమలనగర్"; |
| అమ్మూత్సర్ యొక్క ఆస్తులు వివరములు తెలపండి | SELECT assets FROM branch WHERE<br>branch_name="అమ్మూత్సర్";   |



Query in Telugu language is given as input on which the processing of the algorithm is performed when submit button is presses. The system works by utilizing the use of syntactic knowledge and the knowledge it has been provided about the relevant database. Hence, it is able to implement the Telugu language input to the structure, scope and contents of the database. The program translates the whole query into the standard query language to extract the relevant information from the database. Thus, these products have created a revolution in extracting information from databases. They have discarded the fuss of learning SQL and time is also saved in learning this query language. The snapshot of the user interface for querying the input is shown in the Fig 1 and the result generated after processing the input is shown in the Fig.2.

**IV. CONCLUSION**

The proposed Telugu Language Interface to Database system, for using Banking database, accepts query in Telugu language that is translated into SQL query, by mapping the Telugu language words with the help of database maintained. Then this SQL query is executed on database to provide output to the user. Major components in this system are Tokenizing ,Query Generation, Query Execution and Result Generation.. Our model works well on the queries relating to single table. The future scope of the work is involved with execution of queries relating multiple tables.

**REFERENCES**

[1] AmandeepKaur, Parteek Bhatia, “Punjabi Language Interface to Database” communicated to International journal of computer science, WASET (World Academy of Science and Technology)  
 [2] Androutsopoulos.G.D. Ritchie and Thanisch, “ Natural Language Inerfaces to Databases-An Introduction” Department of Computer Science , University of Edinburgh March 1995.  
 [3] G. Hendrix ESacrdotiD.Sagalowicz and J.Slocum “Developing a Natural Language Interface to Complex Data” ACM Transactions on Database Systems pp:105-147, 1978.  
 [4] 4.RamiReddyNandiReddy, SivajiBandyopadhyay “Dialogue based Question Answering System in Telugu” published at EACL 2006 Workshop on Multilingual Question Answering - MLQA06  
 [5] W.A.woods R.M. Kaplan and B.N Webber “ The LUNAR Sciences Natural Language Information Systems” Cambridge 1972.



Fig. 1 Snapshot of User Interface for querying the Input

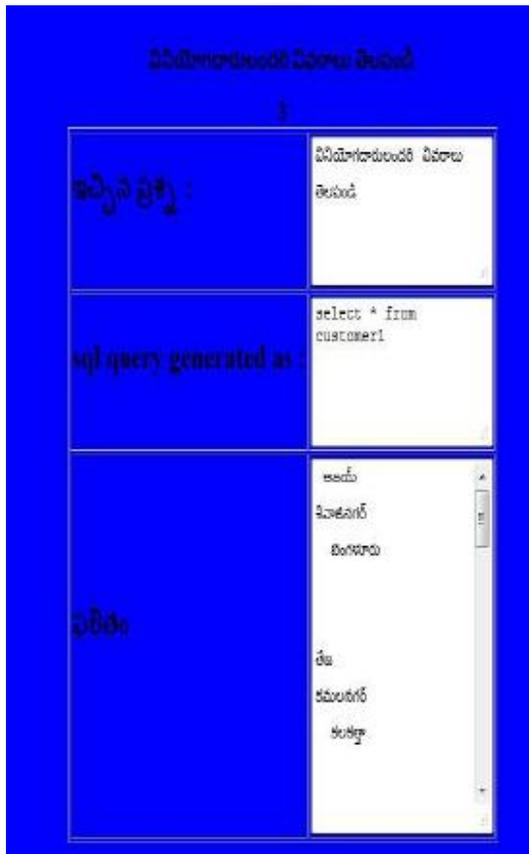


Fig. 2 Result generation

**BIOGRAPHY**



**D.Ramesh** is presently working as Associate Professor in epartment of IT, Jawaharlal Nehru Technological University Hyderabad. His research interests include Databases, Data mining and bject Oriented approaches..



**Dr. Suresh Kumar Sanampudi** is presntly working as Assistant Professor in the department of IT, Jawaharlal Nehru Technological University Hyderabad. His research interests include Artificial Intelligence, Natural Language Processing, Cloud Computing, Algorithms and Information Security