Q&A on Database using NLP

Nikita Mandhan
Department of Computer Engineering
Government College of Engineering, Jalgaon,
Maharashtra, India

Aishwarya Ahuja
Department of Computer Engineering
Government College of Engineering, Jalgaon,
Maharashtra, India

Komal Jain
Department of Computer Engineering
Government College of Engineering, Jalgaon,
Maharashtra, India

H.D.Gadade
Department of Computer Engineering
Government College of Engineering, Jalgaon,
Maharashtra, India

Abstract - As we know today almost every IT application needs database for storing and retrieving the information. For this, it requires to have the knowledge of technical language like SQL. But, it is not compulsory that every person dealing with database should know SQL because they do not belong to the technical background. Querying the database in natural language is a convenient method for data access as compared to the technical languages like MySQL or SQL. So, due to this the development of Natural Language Database Interfaces (NLDBI) has taken place which allows the user to query the database in his natural language. This paper emphasize on the structural design method for translating English Query into SQL using the Data Dictionary provided in the database.

Keywords: SQL, NLDBI, tokens, query, mapping, data dictionary, natural language, database.

I. INTRODUCTION

Natural language processing is becoming one of the most active areas in Human-computer Interaction. The goal of NLP is to enable communication between people and computers without resorting to memorization of complex commands and procedures. In other words, NLP is a technique which can make the computer understand the languages naturally used by humans. While natural language may be the easiest symbol system for people to learn and use, it has proved to be the hardest for a computer to master. Despite the challenges, natural language processing is widely regarded as a promising and critically important endeavor in the field of computer research. The general goal for most computational linguists is to instill the computer with the ability to understand and generate natural language so that eventually people can address their computers through text as though they were addressing another person. The applications that will be possible when NLP capabilities are fully realized are impressive computers would be able to process natural language, translating languages accurately and in real time, or extracting and summarizing information from a variety of data sources, depending on the users requests. This paper describes a natural language database interface that wires complex queries based on a probabilistic context free grammar (PCFG) to relational database. First we summarize some classic NLDBI systems. Consequently we discuss the overall system architecture of the natural language database interface, some implementation details and experimental results[2].

II. RELATED WORK

ELIZA (Joseph Weizenbaum, 1964) is a simulation of a Rogerian psychotherapist, written by Joseph Weizenbaum between 1964 to 1966. Using almost no information about human thought or emotion, ELIZA sometimes provided a startlingly human-like interaction. When the "patient" exceeded the very small knowledge base, ELIZA might provide a generic response. For example, responding to "My head hurts" with "Why do you say your head hurts?[5]

SHRDLU (Terry Winograd, 1968) was an early natural language understanding computer program, developed by Terry Winograd at MIT in 1968–1970. With it, the user carries on a conversation with the computer, moving objects, naming collections and querying the state of a simplified "blocks world", essentially a virtual box filled with different blocks. It was written in the Micro Planner and Lisp programming language on the DEC PDP-6 computer and a DEC graphics terminal. Later additions were made in the computer graphics labs at the University of Utah, adding a full 3D rendering of SHRDLU's "world"[6]
LUNAR (Woods, 1973) involved a system that answered questions about rock samples brought back from the moon. Two databases were used, the chemical analyses and the literature references. The program used an Augmented Transition Network (ATN) parser and Woods' Procedural Semantics. The system was informally demonstrated at the Second Annual Lunar Science Conference in 1971. [1]
LIFER/LADDER (Hendrix, 1978) was one of the first good database NLP systems. It was designed as a natural language interface to a database of information about US Navy ships. This system, as described in a paper by Hendrix, used a semantic grammar to parse questions and query a distributed database. The LIFER/LADDER system could only support simple one-table queries or multiple table queries with easy join conditions. [4]
QUESTION-ANSWERING SYSTEM (Nguyen Tuan Dang, 2009) proposed a method to build a specific Question-Answering system which is integrated with a search system for e-Books in the library. Users can use simple English questions for searching the library with information about the needed e-Books, such as title, author, language, category, publisher… In this research project, the main focus is on fundamental problems in the natural language query processing: approaches of syntax analysis and syntax representation, semantic representation, transformation rules for syntax structure of semantic structure. [7]

III. PROBLEM DESCRIPTION

A huge amount of labour is required if we wish to obtain only the required information from the entire repository of information system. Natural language processing is a process by which the user query (entered in English language) in natural language will be converted to a SQL query based on the query entered as shown in fig1.
Any ordinary person is not expected to know the SQL language, and hence this system would help him in generating the same, so that information retrieval is easier for the database, as database understand the SQL language only.

The objective is to parse the query and with the help of the dictionary, carry out different phases like morphological analysis, syntactic analysis, semantic analysis etc. and finally generate the SQL query.

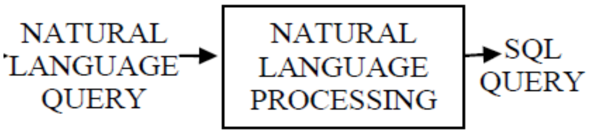


Figure 1.Problem Description

Consider a database, say DB. Within this DB database we have placed certain tables with attributes, which are properly normalized. Now if the user wishes to access the data from the table, he/she has to be technically proficient in the SQL language to make a query to the DB database. Our system eliminates this part and enables the end user to access the tables in his/her language.

Let us take an example:
Suppose if we want to view information about a particular biscuit from the Biscuit table, then we are supposed to use the following query:
SELECT * FROM Biscuit WHERE Biscuit_name=‘<biscuit name >‘;

But a person, who doesn‘t know SQL, will not be able to access the database unless he/she knows the syntax and semantics of firing a query to the database. But using NLP, this task of accessing the database will be much simpler. So the above query will be rewritten using NLP as:
Give the information of the Biscuit whose name is <biscuit name>. Both the SQL statement and NLP statement to access the Biscuit table would result in the same output the only difference being, a normal person who doesn‘t know anything about SQL can easily access the DB database.

IV. SYSTEM DESIGN

We can explain what is the actual process carried out within the Natural Language Processing system by means of a method which is also known as ―Levels of Language also known as Synchronic Model of language. The previous sequential model hypothesis is based on the fact that the levels of Human Language Processing system follow one another in a strict and sequential manner. According to psycholinguistic research, the levels within a language processing interacts in various orders and hence it is much more dynamic.

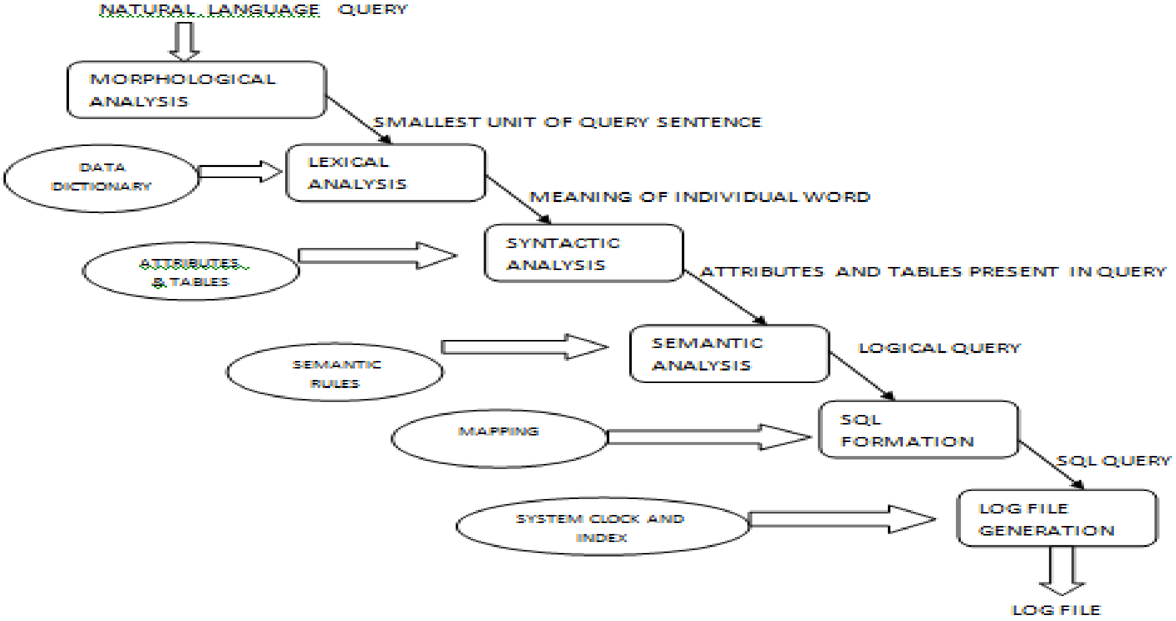


Figure 2: System Design

All the levels shown in fig 2 are explained below:

i. *Morphology*

In this phase, the sentence is broken down into tokens-the smallest unit of meaning. At this level, we split the given input query sentence in natural language into all the words it contains and store the words in a list. For example, if the given input query is ―find cost of the parle biscuit, then in this phase, each word of the sentence, i.e. find , cost, of, the, parle ,biscuit will be stored in a list like [‘find‘, ‘cost‘, ‘of‘, ‘the‘, ‘parle‘, ‘biscuit‘]

ii. *Lexical*

At this level, humans, as well as NLP systems, interpret the meaning of individual words. Each word of the tokenized sentence will be mapped with the meaning of the same word present in the data dictionary. For example, from the list generated in the morphology phase, the words will be mapped as
―find: select,
―cost: cost,
―biscuit: biscuit.

iii. *Syntactic*

At this level at first we find the attributes present in the given input query from the words generated in the lexical phase. Each of them is checked with the attributes in the dictionary which contains all the tables along with their attributes. And then we find the tables which contain the attributes of the given input query. For example, of the output generated in the previous phase, we derive the attributes in the query as ―cost and which belongs to table ―biscuit

iv. *Semantic*

Semantics focuses on the study of meaning of the words present in the natural language query and the relation between signifiers like words, signs, phrases and what do they actually stand for. A field of semantics called Linguistic semantics deals with the study of meaning which interprets human expression through language. This level deals with checking the different conditions like where clause, relational operators, aggregate functions, natural join and build the SQL query accordingly. The SQL query after checking all the conditions is ―select cost from biscuit.

The following Workflow diagram shown in fig 3 below shows the input query with multiple attributes .

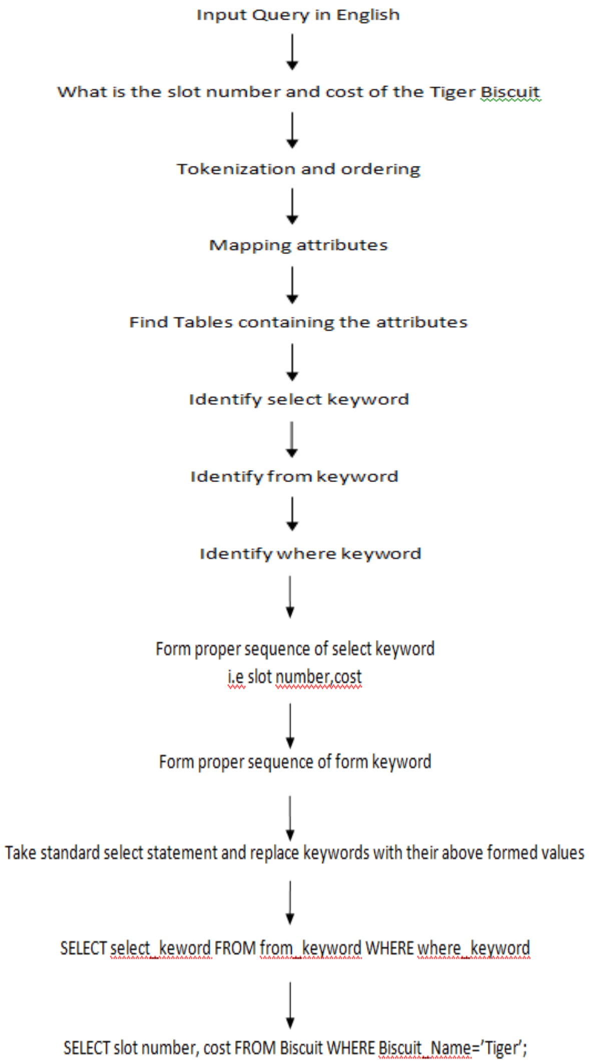

Input Query in English
What is the slot number and cost of the Tiger Biscuit
Tokenization and ordering
Mapping attributes
Find Tables containing the attributes
Identify select keyword
Identify from keyword
Identify where keyword
Form proper sequence of select keyword
i.e slot number,cost
Form proper sequence of form keyword
Take standard select statement and replace keywords with their above formed values
SELECT select_keword FROM from_keyword WHERE where_keyword
SELECT slot number, cost FROM Biscuit WHERE Biscuit_Name='Tiger';

Figure 3.Workflow diagram

ALGORITHM:

- Accept the input from the user in the form of text.
- Split the input query and store it in a list, i.e. tokenize the input sentence.
- Each word of the tokenized sentence will be mapped with the meaning of the same word present in the data dictionary.
- Find all the attributes of all the tables.
- Examine the query and find the table present in the query and the attributes present in the query.
- Find the attributes which belong to table present in the query.
- Find the attribute which do not belong to the table in the query (if any).
- Now find the tables which will contain the pair of ((attribute which do not belong to the table in the query), (other attributes present in the table in the query)).
- Select any one table. Thus we will obtain the tables required for natural join.
- For a natural join query, find out the common attribute of the 2 tables and form the inner query. Then form the outer query according to the different conditions.. Merge both of them and generate the final query.
- For a simple query, generate the final query by checking the different conditions accordingly.
- If there are 2 tables, then perform a natural join on the 2 tables with appropriate attributes of the tables.
- Obtain the conditions of the where clause (single condition or multiple condition by finding the ―and‖ word in the input query), aggregate function (checking whether any aggregate function (like sum, avg, count , etc) present in the query) and the relational operators between the conditions from the list of attributes. Add these to the final query.

- Print the final query.
- Write the generated SQL query in a —log file along with an index number and time of the query using the system clock, so that we can retrieve the query generated at a particular time giving the time as input.

FLOWCHART:

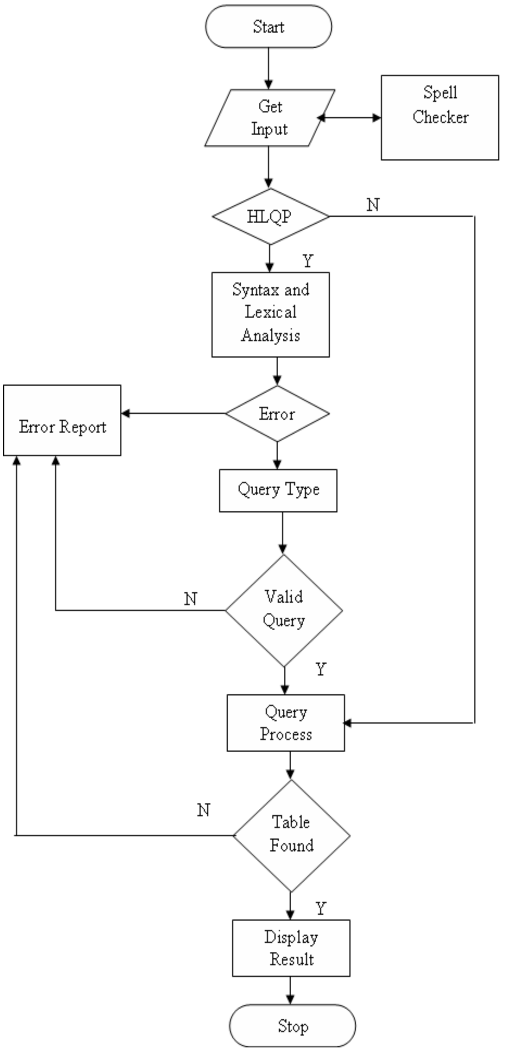
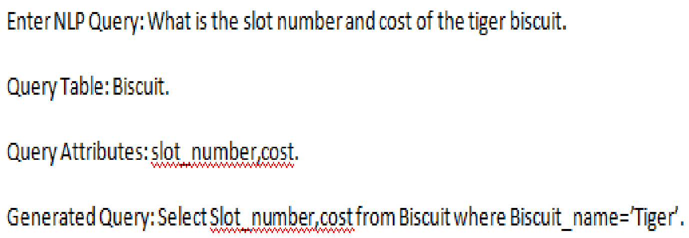


Figure 4: Flowchart

The inputs to the flowchart have both the queries i.e. Normal query and also the human language query. When the input enters, the spell checker will be processed simultaneously. After getting the input it undergoes check for HLQP, if it is false then it is normal query, so it will be passed directly to query process. If it is true then the HLQP is undergone the process of lexical, syntax and semantic analysis. The error handling is done accordingly for all process. After the analysis process it will send to query process, there the query will be processed and then the results will be displayed.

RESULT:



```
Enter NLP Query: What is the slot number and cost of the tiger biscuit.

Query Table: Biscuit.

Query Attributes: slot_number,cost.

Generated Query: Select Slot_number,cost from Biscuit where Biscuit_name='Tiger'.
```

V. CONCLUSION

Natural Language Processing can bring powerful enhancements to virtually any computer program interface. This system is currently capable of handling simple queries with standard join conditions. Because not all forms of SQL queries are supported, further development would be required before the system can be used within NLDBI.

VI. FUTURE WORK

The Future enhancement in this field is basically the Data Dictionary can be extended to a global level. Also, various complex queries could also be generated by improving the pattern recognising and matching skills. Thus, we can improve the answering capability of the system by implementing the future technology.

REFERENCES

[1] Huangi,.Guiang Zangi, phillip C-y Sheu A Natural Language database interface based on probabilistic context free grammar". IEEE international workshop on Semantic Computing and systems 2008

[2] http://www.e-ijaet.org/media/61I8-IJAET0805933-NATURAL-LANGUAGE-QUERY.pdf

[3] www.ijcta.com/documents/volumes/vol3issue1/ijcta2012030153.pdf

[4] Hendrix. G.G, Sacerdoti, E.D,sagalowicz. D. Slocum. J. "'Developing a natural Language interface to complex data in *ACM Transaction on database* system. 3(2). pp. 105- 147,1978.

[5] Weizenbaum, Joseph (January 1966), "ELIZA—A Computer Program For the Study of Natural Language Communication Between Man And Machine", Communications of the ACM 9 (1): 36–45, doi:10.1145/365153.365168.

[6] Terry Winograd, "Procedures as a Representation for Data in a Computer Program for Understanding Natural Language", MIT AI Technical Report 235, February 1971

[7] Nguyen Tuan Dang, and Do Thi Thanh Tuyen. Natural Language Question Answering Model Applied To Document Retrieval System. World Academy of Science, Engineering and Technology 51 2009, pp.36 – 39