International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 05 Issue: 03 | Mar-2018 www.irjet.net p-ISSN: 2395-0072

QUERYING DATABASE USING NATURAL LANGUAGE INTERFACE

Bhaskar Anand[1], Shivani Goyal[2], Osheen Kaul[3], Vaibhav Pandey[4],

Guided By: Prof. Vaishali L. Kohle

[1,2,3,4] *Students, Computer Engineering, D.Y. Patil College of Engineering Akurdi, Pune University Pune, MH, India*
[5] *Professor, Computer Engineering, D.Y. Patil College of Engineering Akurdi, Pune University Pune, MH, India*

Abstract - *The extraction of useful information from a database is one of the most important task performed in an organisation or at a primary level. The demand for increasing technical staff and hence the expenditure weighs heavy on an organisation. This paper aims at creating a model to enhance the interactions that take place between a user and the database. Everyone who might need some information extracted from the database might not be well acquainted with MySQL or similar query languages. This acts like a barrier to both understanding, and profession. The model aims at providing functionality and ease of communication to a system in extraction of information, by allowing the user to communicate in English with the system, and state the query in natural language which is understood by the system and correctly reciprocated. The system has an ability to understand natural language. A data dictionary has been provided to the system and suitable grammar has been used. Multiple steps used in the understanding process are tokenisation, parsing, removing stop words, conversion between lower and upper case alphabets, speech tagging etc. The processed query is converted into a SQL query which is further processed by the system to provide the correct result.*

Key Words: MySQL, Tokenisation, Parsing, Stop Words, SQL query & database.

1. INTRODUCTION

The growth in the amount of data has been exponential in the recent times. The manual evaluation of such data has become impossible in every sense. Querying has become the most basic and important part of any kind of information retrieval. In an organisation, at multiple platforms, data is required by all types of professional posts in an arranged matter. However, it is not necessary that all the employees at all the posts must be mastered in SQL or query processing. This scenario leads for the need for formation of such a system which can understand natural language. To create such a system the basic effort required is the combination of Artificial Intelligence with basic linguistics of a particular language, which may or may not be domain specific. Natural Language Processing (NLP) has therefore emerged as one of the most frequently accessed and most important field of human-computer interaction. The purpose of the paper is to enable people from all academic and non-academic, all different professional backgrounds to access data and acquire the suitable result without the use of any technical or complex query language, and instead get an answer using mono linguistic platforms.

1.1 Existing Systems:

Natural Language Database Interface (NLDBI) has been a topic for intensive research for the past 15 years and multiple architectures and systems have been proposed for the same. Different research works have established different theories and multiple executions of NLDBIs.

1.1.1 LUNAR:

Lunar came to development action in 1971. It solved queries regarding samples of rocks which were brought back to the earth from the moon. Lunar system used two databases which included one for chemical analysis and one was used for literature references. Semantic grammar is used by Ladder for parsing the natural language questions into a query for a distributed database. It provided information about US Navy ships.

1.1.2 LADDER:

Semantic grammar is used by Ladder for parsing the natural language questions into a query for a distributed database. It provided information about US Navy ships.

1.1.3 Chat-80:

Chat-80 was embedded with different facts about multiple countries. A small set of English language words sufficient for querying the database were also added. Chat-80 acted as one of the most sophisticated systems.

2. Proposed System Architecture:

The system leans on an architecture which tries to maximise the interaction between the computer and humans coming from all different academic backgrounds and professions. Considering the specifically responsive structure of databases to only standard queries and SQL and its otherwise unresponsive nature the architecture is framed which contains multiple layers, out of which some are liable to extract data, whereas some are responsible for understanding the nature and the problem specified by the user in natural language. The main three layers of the system are User Interface, Natural Language Processor, and the translator.

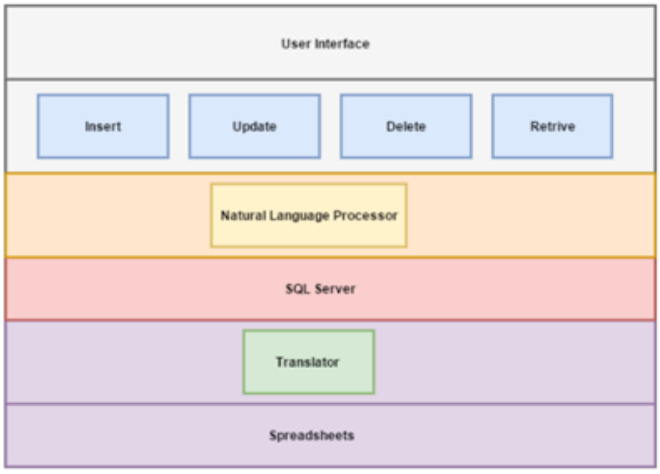

Fig -1: System Architecture.

The User Interface layers consists of an user friendly GUI for the user with multiple options like insert, update, delete etc. The Natural Language Processor consists of the SQL server, and the third layer consists of the database.

A GUI is created for taking input from the user in natural language. This is followed by the linguistic component which advances in three phases.

Phase 1: Morphological analysis: This phase consists of token analysis. Streams of tokens are created from the sentence that has been input by the user through the GUI. The ambiguity of the sentence is minimised in this phase and a spelling checker checks and verifies the spelling.

Phase 2: Syntactic Analysis: In this phase a parse tree is returned. This parse tree shows how word will be related to one another.

Phase 3: Semantic analysis: In this phase the parse tree is assigned with some logical meaning using multiple semantic rules.

Database will be created to interface with the intermediate query that has been generated and the output will be displayed on the GUI.

Database component: This component has two tasks. First, it translates the logical query into a database query, by mapping each element of the intermediate logical query to its corresponding clause in the database query. Further, it displays the answers returned by the DBMS in tabular form. Connection between the modules will be created for implementation.

2.1 Advantages:

Like other systems, NLDBI also has some merits as well as demerits. Advantages are following:

1. No need to know artificial language as there are some languages which we called conventional query languages and are very hard to interpret. In NLDBI, users can use their normal language for querying the database, so no user has to read any kind of language for querying.

2. No need to know the outer structure of data to query in formal language. One should be aware of location of the data where it is stored, but this is not required in NLDBI.

3. Easy to use. For taking data from NLDBI system we require a single input, while a form based may contain multiple input. In case of query language a question needs to be stated by using multiple statements which may consist one or more sub queries with some joint operations.

4. Another merits of NLDBI creation in concern of natural language interface that support anaphoric and elliptical expression. NLDBI of this kind allow the use of brief underspecified questions where the sense of each query is assisted by the discourse context.

5. Easy to use for more than one database tables queries that involve multiple database tables like list the name of the farmers who lost crop worth more than 50000/- during flood are hard to form in graphical user interface as compared to natural language interface.

2.2 Disadvantages:

Many NLDBI systems have been developed so far for business purpose use but use of NLDBI system is not wide spread and it is not the primary choice for interfacing to database. The absence of acceptance is mainly due to disadvantages which are given below:

1. Linguistics coverage is not obvious: At present all NLDBI systems can recognise some subsets of a natural language but it is quite hard to explain these subsets. Even some NLDBI systems cant hold certain query belong to their own subsets. This is not the case of formal language like SQL. Because the formal language description is clear and give the corresponding answers of any statements that observe the given rules.

2. Linguistics vs. imaginary failure: When NLDBI cant interpret a question, it is often not clear to the user whether the refused question is out of systems conceptual analysis or it is outside the systems linguistic analysis. Thus user often try to give input by changing the phrase question referring to concepts the system does not know because they think the problem is caused by the systems restricted linguistic coverage.

3. Inappropriate Medium: It has been stated that natural language is not an appropriate medium for communicating with a computer system. Natural language is claimed to be too wordy or too ambiguous for human-computer interaction. NLDBI users have to type long questions, while in form-based interfaces only fields have to be filled in, and in graphical interfaces most of the work can be done by mouse-clicking.

4. Unrealistic expectations- Mostly people depend on NLDBI systems capability to process a natural language query: they assume that the system is intelligent so it can comprehend facts. Therefore rather than asking precise questions from a database, they may ask questions that involve complex ideas, certain judgments, reasoning capabilities which an NLDBI system cannot be relied upon.

2.3 Speech Conversion modules:

For many years, speech has been considered as a more convenient way of communication and has been preferred over written and textual forms of communication. Speech conversion for distant communication has provided a great aid for the visually challenged and disabled people. The system created has been equipped with a speech conversion module. This is specially embedded to remove the barrier between visually challenged people and the vast world of database. Using this module a question can just be spoken into the system which is in natural language, e.g English. This is processed as an intermediate query which further gets mapped into an SQL query and the correct answer is retrieved. Not only does speech conversion aid the specially abled, also it saves time and provides a better angle of efficiency to the system. A sense of high-tech mechanism is incorporated into the system which makes it more representable in the market and also at multiple professional fronts.

3. CONCLUSIONS

To work with any RDBMS one should know the syntax of the commands of that particular database software (Microsoft SQL, Oracle, etc.). This is the barrier that has been fought by the system being created. Here the Natural language processing is done on English i.e. the input statements have to be in English. Input from the user is taken in the form of questions (wh- form like what, who, where, etc).A limited Data Dictionary is used where all possible words related to a particular system will be included. The Data Dictionary of the system must be regularly updated with words that are specific to the particular system. Ambiguity among the words will be taken care of while processing the natural language. The system can be adopted by multiple business oriented organisation where the extensive task of data analysis can be done by a statistical scientist rather than an engineer. Using this system the precision of visuality and originality of exact thought can be maintained as no middleman is incorporated into the process due to extraction.

REFERENCES

1. Al. GauriRao, Natural language Query Processing Using Semantic Grammar, in International Journal on Computer Science and Engineering, Vol. 02, pp. 219223 , 2010.

2. N. Nihalani, S. Silakari, and M. Motwani , Natural language interface for database: a Brief review, in International Journal of Computer Science Issues 8 (2) p. 600608, 2011.

3. Androutsopoulos, G.D. Ritchie, and P. Thanisch, Natural Language Interfaces to Databases An Introduction, in Journal of Natural Language Engineering 1 p.2981,1995.

4. Y. Li, H. Yang, and H.V. Jagadish, NALIX: an interactive natural language interface for querying XML, in Proceedings of the International Conference on Management of Data, pp. 900902 ,2005.

5. Huangi, GuiangZangi, and Phillip C-Y Sheu, A Natural Language database Interface based on probabilistic context free grammar, in IEEE International workshop on Semantic Computing and Systems, 2008.

6. D.L. Waltz., An English Language Question Answering System for a Large Relational Database, Communications of the ACM, pp. 526 539, July 1978.

7. Hendrix, G.G., Sacerdotal, E.D., Sagalowicz, D., Slocum, J., Developing a natural language interface to complex data, in ACM Transactions on database systems, 3(2), pp. 105147, 1978.

8. Warren, D., Pereira, F. , An efficient and easily adaptable system for interpreting natural language queries, in Computational Linguistics. Vol 8, pp. 34, 1982.

9. Y. Li, H. Yang, and H.V. Jagadish, NALIX: an interactive natural language interface for querying XML, in Proceedings of the International Conference on Management of Data, pp. 900902 ,2005.

10. B.J. Grosz, TEAM: A Transportable Natural-Language Interface System, in Proceedings of the 1st Conference on Applied Natural Language Processing, Santa Monica.