Natural Language Processing Systems In A Smarter Way

Afreen Taj[1], Hemalatha K N[2], Kavya A[3], Rajendra M[4]
[1,2,3,4] Atria institute of technology

Abstract- *Present day databases contain a huge measure of data put away in an organized organization. This data is prepared to get self-learning and the procedure of data extraction from a Database System is for non-master clients as it requires a broad learning of DBMS dialects. Hence, an unavoidable need emerges to connect the hole between client necessities and the arrangement of a basic data recovery framework whereby the job of a specific Database Administrator is repealed. In this paper, we propose a technique for structure an Intelligent Querying System (IQS) by which a client can fire inquiries in his own (characteristic) language. The framework initially parses the info sentences and afterward creates SQL questions from the characteristic language articulations of the info. These inquiries are thus mapped with the ideal data to create the required yield. Consequently, it makes the data recovery process straightforward, powerful and solid.*

Keywords- SQL; Natural Language Processing; NLIDB; Data Warehousing; Data Mining; Query Generator; MR Generator; Semantic Builder.

I. INTRODUCTION

Databases are built with the aim to facilitate activities like data storage, processing, and retrieval associated with data management in information systems. In relational databases, to extract (retrieve) information, the query is formulated in such a way that the computer can analyze it and produce the output desired [1]. The Structured Query Language (SQL) rules are generally used in all languages used for relational database systems. The SQL rules are based on a Boolean interpretation of the queries. In recent times, there is an ascending demand for non-skilled users to extract information from a database without being mandated to learn standard query languages. This will help bridge the communiqué gap between users and information storing databases [2].

One preferred standpoint of NLIDB (Natural Language Interface to Database) should be that clients are not required to gain proficiency with any counterfeit correspondence language. In any case, graphical interfaces and structure-based interfaces are simpler to use by infrequent clients; stills, summoning shapes, connecting outlines, choosing limitations from menus, and so forth establish counterfeit correspondence dialects that must be scholarly and aced by the end-client. Conversely, a perfect NLIDB would enable inquiries to be planned in client's local language. This implies that a perfect NLIDB would be progressively reasonable for incidental clients as there would be no requirement for them to spend time in learning the framework's correspondence language.

This has a major advantage because file system was used to store small chunks of data whereas nowadays people need databases to store lumpsum amount of data and with this concept everyone, even from non IT background can use IQS for easier access, processing, storing of data.

II. OVERVIEW

Since databases have complex structures, it is troublesome for a layman to perform activities on them. In this manner, we propose a framework in which client can enter common language questions according to prerequisite. A social database the executives for example RDBMS is a program that gives us a chance to make; update; and regulate the social database. RDBMS economically utilizes SQL to get to the database. It comprises of data about trait names, information types, table structures and table connections. To extricate data from the social database, a semantic manufacturer is utilized. Phonetic portrayal of the arrangement of database traits with every single imaginable equivalent word is known as dictionary. To produce a vocabulary, database components are removed and after that these components are part into individual words. WordNet is used to recognize the possible synonym for given set of words [7].

Semantic builder works intelligently to extract and process the information so as to intensify the semantic map. For a particular database, only one semantic map has to be created. The semantic map consists of lexicon and information about the relational database like tables and their relationships [3]. Automatic Semantic Map Generator can be used for automatic mapping of the database which can reduce the user's effort of creating the semantic map manually. It can also solve the ambiguity problems for which Intelligent Engine is used. Fig. 1, shows an example of parse tree after parsing and tokenization. In the second stage, NLIDB system requires the translation of queries into an intermediate code.

This intermediate code is termed as Meaningful Representation. MR Generator takes tokens and generates corresponding MR map. Once MR map is created, MR checker evaluates and verifies its correctness [4]. In the last stage, IQS produces the SQL query and represents the extracted information in a meaningful structure like graph, table, etc.

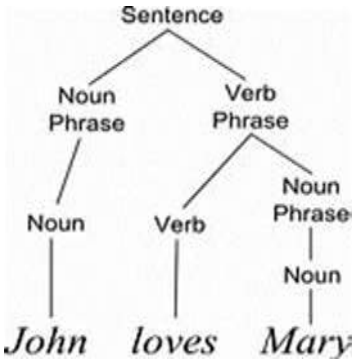

Fig. 1. Parsing of sentence

III. EXISTING SYSTEM

The standard way to deal with Natural Language Interface to Database Systems depends on 'Semantic Analysis'. In this,the semantic sentence structure is utilized to parse the characteristic language question input given by the client. This methodology makes a semantic language structure for every database and utilizations it to parse the regular language question. The semantic punctuation makes a portrayal of the semantics (or importance) of the sentence. After the semantic portrayal investigation, a comparing database question can be created in any database language. At that point, according to necessity, a reaction is produced or proper move is made [6].

The benefit of this methodology is that we get customized language for every database. There could be programmed age of a NLP framework for databases be that as it may, in practically all cases, the database does not have adequate data to make a dependable NLP framework [5]. To deal with all conceivable questions, extra data about what the information in the database speaks to, should be given to make NLP frameworks.

IV. PROPOSED SYSTEM

A. Objective

The purpose of IQS is to provide data processing power to non-technical personnel, who constantly rely on Database Administrators to process data. This requires the data to be processed using Natural Language Processing.

B. Project Scope

The scope of the system is as follows:

- To work with RDBMS, one should know the syntax of the commands of that particular database software (Oracle, Microsoft SQL, etc.).
- Here, the Natural Language Processing is done in English i.e. the input statements have to be in English.
- User input is entered in an interrogative format- WHat, WHo, WHere, WHen and WHy.

The system proposed will include include the following modules:

- GUI: Designing the Front-End or the User Interface where the user will enter the query in Natural Language.
- Parsing: The program, usually part of a compiler that receives input in the form of sequential source program instructions, interactive online commands, markup tags or some other defined interfaces; and breaks them into tokens.
- Query Generation: Once the user statements are successfully parsed, the query is generated in SQL and is given to the back-end database.
- Data Maneuvering

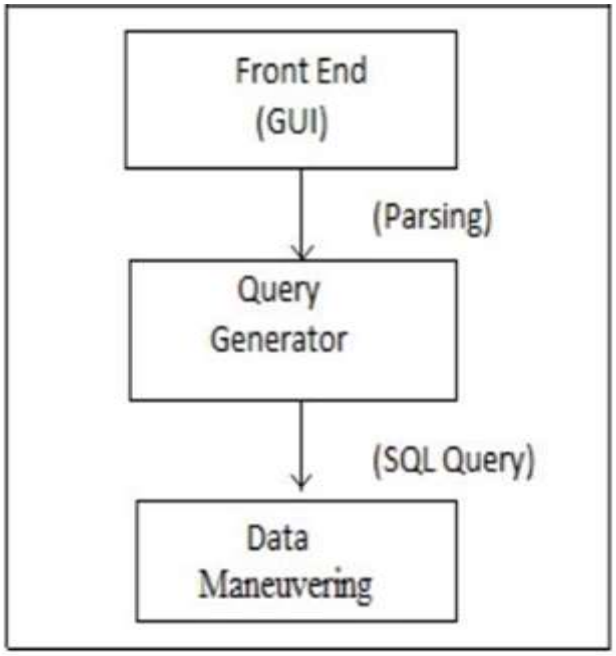

Fig. 2. Different Modules of Proposed System

V. SYSTEM ARCHITECTURE

IQS is composed of different components having distinct features which are integrated together in a comprehensive manner. Architecture of IQS can be divided into three phases- Semantic Building Phase, MR Generation Phase and Query Generation Phase.

A. Semantic Building Phase

Semantic refers to the study of algorithms which categorizes the phrases, words and morphemes that form a sentence. The process by which such representations are created and mapped to linguistic inputs is termed as Semantic Analysis. Semantic Builder generates the Semantic Map by extracting the information by performing operations like type checking and type conversion on the input stream. The working semantic builder depends upon the complexity of attribute naming.

B. MR Generation Phase

The main objective of IQS system is to convert natural language inputs to SQL queries. However, it is a complicated multi-phased process which first requires the input to be represented in an intermediate form. This intermediate form is termed as meaningful representation or MR. It is basically used to show the relationships between different tokens.

Tokenization:

The tokenizer is used to break up the stream of inputs into words, symbols or other meaningful elements called tokens. The natural language input of user can be broken down into one or more tokens. Tokens can be, thus, classified into three types: reserved keywords like what, where, how many, etc., attributes like f_name, dept_no, emp_id, etc., and alpha numeric values. Tokenization is necessary to find the context of the input query of the user w.r.t. the database

The tokenization process works in the following way:

1. Input tokens should match with at least one database element.
2. Each attribute token should map with a value token.
3. Finally, either a value or an attribute token should match with the relation token.

C. Query Generation Phase

This is the final phase of the system where SQL queries are generated by mapping MR to Semantic representation. WordNet is used in this phase to identify the synonyms of attributes. For, e.g., {entire, faculty} and {all, professors} will produce an expanded set of {all, teachers} in which teachers will be mapped with the database attribute "teachers" and "all" will be taken as a reserved keyword.

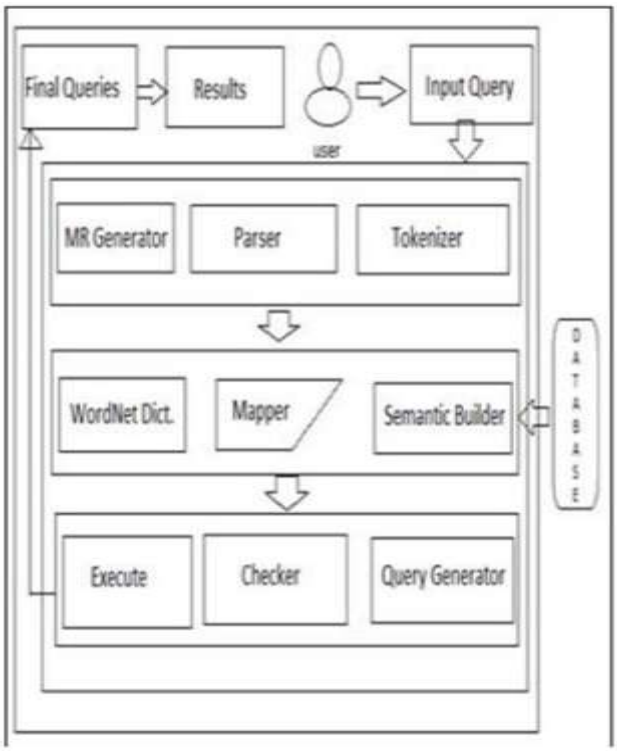

Fig. 3. System Architecture

VI. CONCLUSION

This framework displays a realistic answer for non-master clients to question a database as normal language questions. IQS shows an abnormal state of productivity in producing exact and precise questions. This is conceivable since it capably makes an interpretation of immovable questions to halfway tractable or totally tractable structures. It moreover empowers the clients to get to the data in generally practical structure by making custom fitted perspectives on the outcomes. It can speak to data as tables, diagrams and pie outlines. In future, IQS can be coordinated with cutting edge information cooperation apparatuses, multi-lingual NLIDBs and profound learning strategies to extricate higher throughput from the framework.

REFERENCES

[1] Khan Tabrez, Shaikh Shagufta, Shaikh Sharmeen, Momin Ummiya,Shaikh Rameeza "NLP to Create SQL Query", IJSRD - International Journal of Scientific Research & Development| Vol. 3, Issue 09, 2015 | ISSN (online): 2321-0613a.

[2] Gauri Rao, Chanchal Agarwal, Snehal Chaudhry, Nikita Kulkarni, Dr S.H. Patil "Natural Language Query Processing Using Semantic Grammar", (IJCSE) International Journal of Computer Science and Engineering Vol. 02, No. 02, 2010, 219-223.
[3] "Surface Mount Technology," [Online]. Available: https://en.wikipedia.org/wiki/Surfacemount_technology.
[4] Ana-Maria Popescu, Oren Etzioni and Henry Kautz, "Toward a Theory of Natural Language Interfaces to Databases," in IUI Proceedings 2003.
[5] Yuk Wah Wong and Raymond J. Mooney, "Learning for Semantic Parsing with Statistical Machine Translation," in Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL2006), NY, 2006, pp. 439-446.
[6] Hendrix, G.G., Sacerdoti, 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.
[7] Seymour Knowles, Tanja Mitrovic "A Natural Language Database Interface For SQL-Tutor", in ANL 1999.
[8] George A. Miller "WordNet: A Lexical Database for English", in Communications of the ACM November 1995/Vol. 38, No. 11.