International Journal of Scientific & Engineering Research Volume 11, Issue 3, March-2020
ISSN 2229-5518

Natural Language Query Processing for SQL And NOSQL Queries

Abhishek Kerkar
Dept. of IT Engineering
PHCET
Rasayani, Panvel

Haider Ali Shaikh
Dept. of IT Engineering
PHCET
Rasayani, Panvel

Prathamesh Satam
Dept. of IT Engineering
PHCET
Rasayani, Panvel

Dr. J.E. Nalavade
Dept. of IT Engineering
PHCET
Rasayani, Panvel

Abstract: **This document present an approach to convert Natural Language Query to SQL Query and NoSQL query effectively. Query Language is a tool for managing data held in a database management system. To retrieve or manage data user have to enter the correct SQL and NoSQL Query. But the users who lacks any knowledge about SQL and NoSQL are unable to retrieve the required data. To overcome this, we proposed a system in Natural Language Processing for converting the Natural Language Query to SQL and NoSQL query. This helps no user to get required information without knowing any complex details about SQL and NoSQL.**

INTRODUCTION

Databases have applications virtually altogether info systems like transport information system, financial information system, human resource management system etc. The continues increase within the size of information and quality within the relation among the entities have resulted in sophisticated SQL and NoSQL question that are terribly difficult to write for an ordinary user. It additionally needs user to understand details of information such as relations, entities and object notation etc. The main problem is that the users who want to get information from the information, doesn't grasp formal languages like SQL and NoSQL.

LITERATURE REVIEW

Query Language is a tool for managing data held in a database management system. To retrieve or manage knowledge user need to enter the proper SQL and NoSQL queries. But the users who don't have any knowledge about Database query language are unable to retrieve the required data. To overcome this, we have a tendency to plan a model in natural language process for changing the tongue question to SQL question. This helps novice user to urge needed content while not knowing any advanced details regarding SQL. Technologies used are Natural Language Query, Natural Language Processing, Speech-to-Text, SQL, Syntactic, Semantic, and Data Dictionary.

EXISTING SYSTEM

A. Tokenization

System will perform tokenization on the entered query by separating it into single words. Each word represents a token. Then these words will be stored in a separate list and passed to Lexical Analyzer.

B. Lexical Analysis

The tokenized list will be mapped with the dictionary. These words will get replaced by the database words from the dictionary and passed to syntactic analysis.

C. Syntactic Analysis

In this step dictionary of table names, attributes and keywords are maintained. Each tokenized word gets mapped with attributes in the dictionary. It is passed to Semantic Analysis for further processing.

D. Semantic Analysis

System will find words which represent conditions or symbols and that word will get mapped with the dictionary. (For Example: If there is "less than or equal to" in the query, it will get mapped with the symbol "<=").

PROPOSED SYSTEM

SQL queries a database is a computer device that stores a set of information. In a relational database, information is stored as arrays, called tables. A relational database can have one or more tables, linked or not to each other. Inputs (information) are gathered in what are called columns (or fields). A group of columns relating to the same entity (object) forms uneatable. A schema, also called a data model.

NoSQL are database that just not only stores data in tabular form but also has certain functionalities. NoSQL was developed to tackle modern technologies requirements. NoSQL mostly use to retrieve or store data in big data, e.g. Hadoop, etc. some database based on NoSQL are mongo DB, Cassandra.

1. A module is a separate unit of software or hardware. Typical characteristics of modular components include portability, which allows them to be used in a variety of systems, and interoperability, which allows them to function with the components of other systems.

2. NLTK system: Actually converts the natural language sentences to SQL and NoSQL query and display the results.

3. Database: Consist of the records that are to be retrieved by NLTK system.

ALGORITHM

[1] The first step in query formation is to process the input query, i.e. divide the user's query into tokens, which are individual words in the query.

[2] Then replace the starting of the query with appropriate syntax. Identify the words in the user's query which are column names (attributes), the words which represent the name of the tables and values, if any, present in the user's query.

[3] Replace synonyms of column names or table names with the actual attributes or table names.

[4] Construct the query as follows:

• Identify the attributes which the user wants to retrieve. This will be appended to the select keyword.

• Identify the table to which these attributes belong. This will be appended to the form keyword.

• Identify the conditions or values, if any, specified by the user in his/her query.

[5] Generate the final query and fire it on the database to get the required result which will be displayed to the user.

SYSTEM ARCHITECTURE

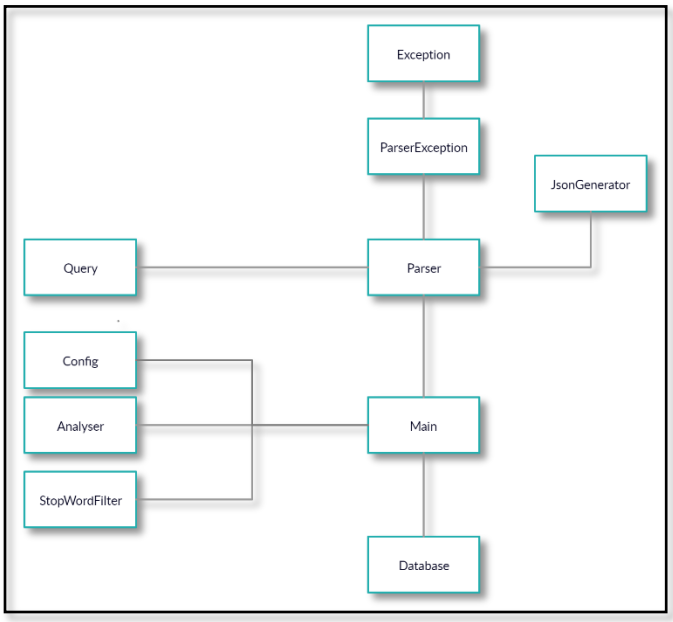

Figure 1. System architecture.

The following figure shows various components of our system.

- Main: - It has various function call, which are required by program.
- Config: - It is used to load csv file which consist of language dictionary. This file is used by Analyser.
- Analyser: - Various functions are performed in this components, Function such as Lexical Analysis, Syntactic Analysis, etc.
- StopWordFilter: - It is to separate user input data word for word. e.g. "it is raining today" this sentence will be separated as "it", "is", "raining" and "today".
- Database: - It is used to store the data in SQL or NoSQL format.
- Parser: - Generation of SQL and NoSQL query is done in this component.
- Query: - Used to display generated query.
- JsonGenerator: - Most of the NoSQL database use Json like Formant for Querying database. Hence JsonGenerator is used to generate such query of NoSQL database.

CONCLUSION

Although we have not been able to clearly compare, our method shows results globally equivalent to most current applications with nevertheless a weakness at the joints .We also note the impossibility of managing silent constraints and generating nested queries. In future works, we plan to deal with

mute constraints, keeping verbs as key words and adding some rules in the grammar. This will, for example, make it clear that "the student named John" means the same thing as "the student whose name is John" or that "the 18-year-old student" equals "the student being 18 years old ". In addition, it is intended to detect the language of the request entered by the user to use a dictionary of synonyms relating to this language and adjust the rules according to the language to make the system robust to others In conclusion, although perfectible, this approach makes it possible to query any SQL database, thus meeting the fixed portability objectives, while maintaining average performance of already existing applications and covering a wide range of selection operations

REFERENCES

- M. Uma, V. Sneha, G. Sneha, J. Bhuvana and B. Bharathi, “Formation of SQL from Natural Language Query using NLP”, International Conference on Computational Intelligence in Data Science (ICCIDS), Chennai, India, pp. 1-5,2019.
- G. J. D. R. Hains, Y. Khmelevsky and T. Tachon, “From natural language to graph queries”, IEEE Canadian Conference of Electrical and Computer Engineering (CCECE), Edmonton, AB, Canada, 2019, pp. 1-4,2019.
- E. Varsha and P. C. Rafeeque, “Translating Natural Language Sentences into Database Query”, International Conference on Emerging Trends and Innovations In Engineering And Technological Research (ICETIETR), Ernakulam, pp. 1-6, 2018.
- C. Baik, H. V. Jagadish and Y. Li, "Bridging the Semantic Gap with SQL Query Logs in Natural Language Interfaces to Databases," IEEE 35th International Conference on Data Engineering (ICDE), Macao, Macao, 2019 pp. 374-385, 2019.
- K. Ahkouk, M. Machkour, M. Ennaji, B. Erraha and J. Antari, "Comparative study of existing approaches on the Task of Natural Language to Database Language," 2019 International Conference of Computer Science and Renewable Energies (ICCSRE), Agadir, Morocco, pp. 1-6,2019.
- A. Kate, S. Kamble, A. Bodkhe and M. Joshi, “Conversion of Natural Language Query to SQL Query”, Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, pp. 488-491, 2018.