iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** V **Month of publication:** May 2023

DOI: https://doi.org/10.22214/ijraset.2023.53426

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue V May 2023- Available at www.ijraset.com

Conversion of NLP to SQL Using Machine Learning Approach

Ms. Sneha. A. Khaire[1], Faisal Sayyed[2], Sudeep Mishra[3], Kanchan Patel[4]
[1]*assistant Professor,* [2, 3, 4]*UG Scholar, Information Technology Engineering, Sandip Institute of Technology and Research Centre, Nashik, India*

Abstract: ***Now days data is increasing rapidly. There are so many new database tools and technologies are rowing, therefore we can store large data, but the problem is that the technology or an interface which can process data and display the data as per the user request is not familiarized with many of the people. It means many people don't have proper knowledge of handling database.***
Actually, NLP to SQL conversion is part of machine learning. The main idea of NLP to SQL conversion is to generate SQL query from natural language. This technique is useful for accessing data from database without having prior knowledge of SQL. This technique can be used by many common people. In this system input is simple English text and query is generated by using POS tagger in python. In this project we are going to implement a system which will generate SQL query from natural language.

I. INTRODUCTION

Storage of data is a crucial task in today's commercial system especially social media, database size is increased and accessing data from database become more crucial part in the recent research world. So many new database tools and technologies are growing, therefore we can store large data, but the problem is that the technology or an interface which can process data and display the data as per the request is not familiarized with many of the people. Most of the businesses and social sites need these types of applications by using the SQL language.
Natural language processing (NLP) is becoming most active techniques to process on human language. In case of social media, the query conversion is very crucial task in terms of getting exact data which is requested by the users. The query or request can be of simple English language statement such as blog, comment, tweets etc., these statements must be converted into proper SQL statement so that exact data can be fetch from database. so, these factors are acting as a precious evidence for implementing the proposed work through this article. The objective of NLP is to facilitate communication among human and computers without multifaceted instructions and procedures. In other words, NLP is the technique that can used the natural languages used by users. An end user can be easily processing their query without knowledge of SQL.
Therefore, in this work the development of system for people to interact with the database in simple English language is implemented and analyzed for the accuracy. This enables a user to input their queries in simple English and get the answer in same language which is referred as Natural Language Interface to a Database (NLIDB) The knowledge extraction is enabled with the successful implementation of SQL generation from the natural language statement.

II. LITERATURE SURVEY

There have been a large amount of research works introducing the theories and applications of Natural Language Interface to a Database (NLIDB). Asking question to databases in natural language is very appropriate and easy method of information access especially for informally users who do not comprehend complex database query language. In fact, database NLP may be one of the most significant successes in NLP since it began. Asking questions to databases in natural language instead of the database complex queries.

1) In 1972, W.A. Woods developed a system that provided a search interface for the database system that stored information about the rock samples that were brought from the moon for research. This system used two databases, the chemical analyses and the literature references. This system used Augmented Transit Network (ATN) parser and semantics of Woods. This system was demonstrated informally at Second Annual Lunar Science conference in 1971.
2) Lifer/Ladder system (1978) was one of the good search interface techniques (i.e. NLP system) which used semantic grammar for parsing the input query and the query generated was given as input on a distributed database system. This system supports single table queries and simple join queries in case of multiple tables.

©IJRASET: All Rights are Reserved

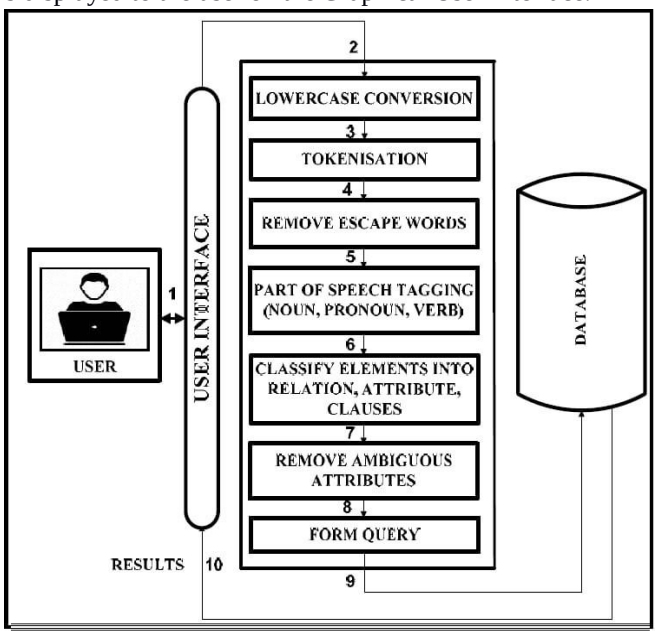
3) Akshay et al. proposed a system which provides a search interface for the users to pose questions in their natural language. The primary goal of this system is to generate a database language query from a NL query. This system includes an additional feature of eliminating spelling errors from user queries and used Word Pair Mining Technique for the same. Then the query in English is mapped to an equivalent SQL query.
4) Prasun Kanti et al. has proposed a system for interfacing a college database that transforms English query to SQL using semantic grammar. The system goes through the morphological, syntactic, semantic phases. The user may ask the question in speech format which is then converted to text using Scripting Language for Android (SL4A). The natural language query is then parsed using parser. A data dictionary stores all the attributes and tables of the database. The attribute identifier then finds out the attributes that are present in the natural language query. With the identified attributes, SQL query is generated.
5) K. Javubar et al. has proposed a user-friendly interface for accessing data from various web sources such as Facebook, Twitter, etc. The architectural layout consists of tokenizing, stemming, parsing and mapping stages. The input natural language query initially undergoes morphological analysis then semantic analysis which is followed by a mapping phase. The three main keywords SELECT, FROM and TO are looked for in the input query. Once these are found, the SQL query is formed.
6) Weaver and Booth implementing the first natural language program on machine translation to crack codes during World War II.
7) After that, most of the systems that were created from this perspective were based on searching in the dictionary for the appropriate words for translation and rearranging words to suit the rules of word order of the synonymous language. This was carried out without looking at the lexical ambiguity inherent in the natural language, as it led to inaccurate and bad results.
8) Applications of NLP had developed dramatically, as rhetorical documents were used to create response-generator text meta descriptions such as McKeown's discourse planner, TEXT , and McDonald's response generator, MUMMBLE.
9) By the 1980s, the concept of natural language had expanded and there was a growing realization of the need to find solutions to the limitations of natural language programming and a general push towards applications that worked with the language in a broad real- world context. Natural language programming grew rapidly from that time until it underwent a major transformation in the early 1990s with the transition to relying on empirical methodologies versus the introspective generalizations that characterized the Chomsky era which had an impact on theoretical linguistics.
10) A study entitled "Text-to-SQL Generation for Question Answering on Electronic Medical Records". aimed to provide services related to health care that are asked by patients in the form of queries through databases, so that these questions are translated into medical inquiries, and then, responses are made from medical records entered into the databases, where the questions are related to several tables, which requires complexity in query strings that may produce false results.
11) The authors of modeled SQL query logs as a query part diagram to im- prove the ability of language interfaces, access information within log information pipes, match words and terms used by the user, and enter them through the system according to their proposed NLIDBS system to enhance the performance of language interfaces with the limitations of poor accuracy in converting NLQ to SQL, a bad effect of user sessions in an SQL query log, data matching confusion, and no ways were found to improve existing deep learning from start to finish.

III. PROJECT IMPLEMENTATION

A module is a collection of source files and build settings that allow you to divide your project into discrete units of functionality. Your project can have one or many modules, and one module may use another module as a dependency. You can independently build, test, and debug each module.

1) *User Interface:* The user interacts with the system via Graphical User Interface and types his/her Natural Language Query for the further output.
2) *Lowercase Conversion:* The Natural Language Query is then translated into lowercase and passed to the tokenization.
3) *Tokenization:* The query after lowercase conversion is then transformed into stream of tokens and a token id is providing to each word of NLQ.
4) *Escape Word Removal:* The extra/stop words are removed which are not needed in the analysis of query.
5) *Part of Speech Tagger:* The tokens are then classified into nouns, pronouns, verb and string/integer variables.
6) *Relations-Attributes Clauses Identifier:* Now the system classifies the tokens into relations, attributes and clauses on the basis of tagged elements and also separates the Integer and String values to form clauses.
7) *Ambiguity Removal:* It removes all the ambiguous characteristics that exists in multiple relation with the same attribute name and maps it with the correct relation.
8) *Query Formation:* After the relations, attributes and clauses are extracted, the final query is built.

9) Query Execution and Data Fetching: The query is then executed and data is got from the database.
10) Results: The final query result is displayed to the user on the Graphical User Interface.



IV. SYSTEM REQUIREMENTS

A. Database Requirements
1) SQLite Version 3.40.0

B. Software Requirements
1) Core i3 processor
2) Windows 10
3) Python Libraries: Pandas, Matplotlib, Numpy

C. Hardware Requirements
1) Desktop/Laptop
2) 6 GB RAM, 1 TB Hard disk

V. TOOLS AND TECHNOLOGIES USED

A. Python

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

Often, programmers fall in love with Python because of the increased productivity it provides. Since there is no compilation step, the edit-test-debug cycle is incredibly fast. Debugging Python programs is easy: a bug or bad input will never cause a segmentation fault. Instead, when the interpreter discovers an error, it raises an exception. When the program doesn't catch the exception, the interpreter prints a stack trace. A source level debugger allows inspection of local and global variables, evaluation of arbitrary expressions, setting breakpoints, stepping through the code a line at a time, and so on. The debugger is written in Python itself, testifying to Python's introspective power. On the other hand, often the quickest way to debug a program is to add a few print statements to the source: the fast edit-test-debug cycle makes this simple approach very effective.

B. NLTK Python Library

NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum. Thanks to a hands-on guide introducing programming fundamentals alongside topics in computational linguistics, plus comprehensive API documentation, NLTK is suitable for linguists, engineers, students, educators, researchers, and industry users alike. NLTK is available for Windows, Mac OS X, and Linux. Best of all, NLTK is a free, open source, community-driven project. NLTK has been called “a wonderful tool for teaching, and working in, computational linguistics using Python,” and “an amazing library to play with natural language.”

C. Machine Learning

Machine Learning is the ability of the computer to learn without being explicitly programmed. In layman’s terms, it can be described as automating the learning process of computers based on their experiences without any human assistance. Machine learning is actively used in our daily life and perhaps in more places than one would expect. Machine Learning is making the computer learn from studying data and statistics. Machine Learning is a step into the direction of artificial intelligence (AI). Machine Learning is a program that analyses data and learns to predict the outcome.

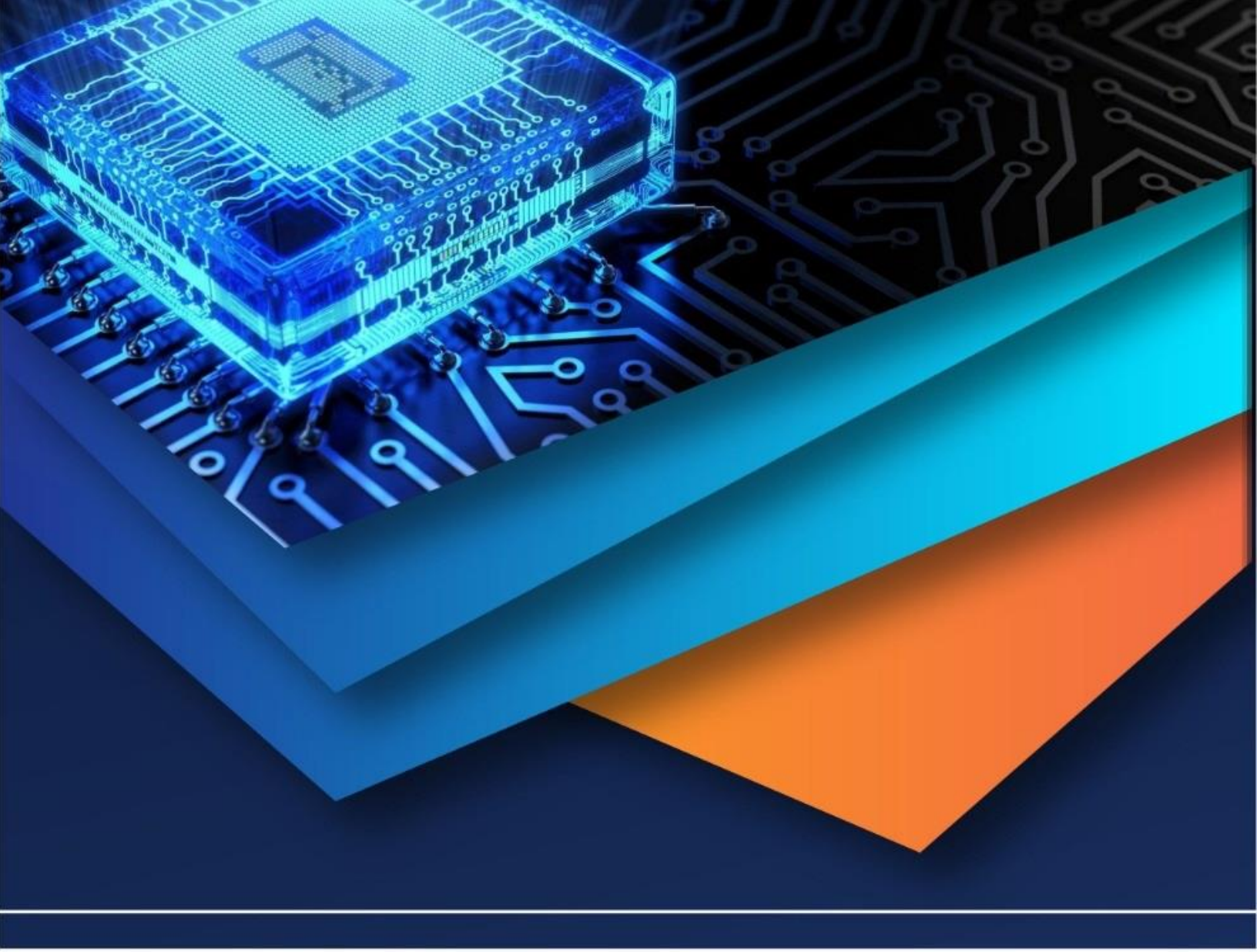
VI. CONCLUSION

Natural Language Processing can bring commanding enhancements to virtually any computer program. Retrieving data from the database requires knowledge of technical languages like SQL. In this project we consider a lightweight approach of translating English queries into equivalent SQL queries. In this approach we look at extracting certain keywords and indicators from an English query written using POS tagger method, and then using a system to generate the query based on the key.

REFERENCES

[1] Natural Language to SQL Conversion System, International Journal of Computer Science Engineering and Information Technology Research (IJCSEITR) Vol. 3 Issue 2, June 2013, 161-166.
[2] Automatic SQL Query Formation from Natural Language Query, International Journal of Computer Applications.
[3] Huangi,GuiangZangi, Phillip C-Y Sheu ―A Natural Language database Interface based on probabilistic context free grammar, IEEE International workshop on Semantic Computing and Systems 2008.
[4] A Survey of Natural Language Query Builder Interface to Database, International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 5 Issue 4, 2015.
[5] Gauri Rao(IJCSE) International Journal on Computer Science and Engineering
[6] TO THEXANDER R., R U KS HA N P. & MAHESAN S. (2013). Natural Language Web Interface for Database (NLWIDB).
[7] Faraj A. El-Mouadib, Zakaria Suliman Zubi, Ahmd A. Almagrous, "Generic interactive natural language interface to databases (GINLIDB)", EC'09 Proceedings of the 10th WSEAS international conference on evolutionary computing, ISBN: 978-960-474-067-3
[8] Rani Nelken, Nissim Francez, "Querying temporal databases using controlled natural language", Proceeding COLING '00 Proceedings of the 18th conference on Computational linguistics - Volume 2
[9] Alessandra Giordani, Alessandro Moschitti, "Semantic mapping between natural language questions and SQL queries via syntactic pairing", Proceeding NLDB'09 Proceedings of the 14th international conference on Applications of Natural Language to Information Systems, ISBN:3- 642-12549-2 978-3-642-12549-2
[10] Alessandra Giordani and Alessandro Moschitti, "Translating Questions to SQL Queries with Generative Parsers Discriminatively Reranked", Proceedings of COLING 2012: Posters, pages 401–410, COLING 2012, Mumbai, December 2012.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)