Support System for creating SQL Queries using Natural Language

Mumuksha Pant
Software Engineering
SRM Institute of Science and Technology
Chennai , India

Vinayak Mishra
Software Engineering
SRM Institute of Science and Technology
Chennai , India

B.Jothi
Software Engineering
SRM Institute of Science and Technology
Chennai , India

Abstract **- Nowadays, all the people have their own personal devices that are connected to the web. Each one of these people try to get the knowledge that they need via the internet. Much of the knowledge is within some sort of database. A person that needs to retrieve the data via a database and has very little or no knowledge about the database languages faces a problematic scenario. Therefore, there is requirement for a means and a system that permits the average person to access natural language within a table. This paper looks to make a system like the one mentioned using NLP by giving natural language question as input and outputting SQL query, to access the related info via a Vehicle database table with no difficulty. The various steps that are there during these processes are lemmatization, tokenization, parts of speech tagging, chunking and mapping the values. The dataset being used to implement the proposed system has a group of structured natural questions on car details. This research would be an overall view of the usage of Natural Language Processing (NLP) and use of regular expressions to map the query in English to SQL.**

I. INTRODUCTION

Among the quick technologically moving world, it has become very important for people to communicate with the computers to get help and assistance in many professional fields like doctors, teachers, Engineers, etc. Attainment of the specified data via the database may be a strenuous task. so as to get the data via the database, the person to have a previous knowledge of Database management System (DBMS) is a must, Therefore a person with no technical background faces difficulty in attaining the information. To look for an answer for such situation and aid human interaction with computers, Natural Language Processing(NLP) techniques are used. Natural Language Processing has been applied to various sectors like Machine Translation where machine understands the words and translates it to another language. Another important application of NLP is chatbot (Chat Robot) which will be used for voice or textual interactions. Our research is focused on Vehicle Data System, where a person can enquire about cars and various details about them. The main goal of this project is to

convert a Natural Language query into a SQL to simplify data extraction.

II. LITERATURE SURVEY

In the year 1972, William Aaron Woods, a renowned researcher in the natural language processing field, created such a system that people to search a database system that that stored data about the rock samples that were brought via the moon for research. This system made use of the Augmented Transit Network parser and Semantics of woods. Prasun Kanti proposed such a system that allowed user to interact with a college database and transformed natural language to sql queries. This proposed system could take input a question in speech format as input and then converted it to text using Scripting for android. In 2014, K. Javubar proposed such a system that allowed a user to access information via Web sources like Facebook, Google, Twitter. The system consisted of different stages such as tokenizing, stemming, parsing and mapping .The input natural language query initially undergoes morphological analysis then semantic analysis which is followed by a mapping phase. There are three main keywords SELECT, TO, FROM that are looked for in the input query which is entered by the user in plain English language. Once these key words are identified, they are placed into the sql query and the query is formed.

III. PROPOSED SYSTEM

Attaining the specified data via a database is sort of problematic for a commoner and needs tons of effort that requires the knowledge of the Database Management Systems and how to work on them. DBMS are not able to handle queries framed in the other languages aside from the usual database languages. So, as to make it easier and interactive for a commoner, our proposed system facilitates a way through which a person can to ask a question in English, which can be processed by several modules to make the same SQL query.

A. An Overview of Query Formation

The user of the system submits an English query in the textual form , which is then sent to several Python modules in Natural Language Tool Kit , also known as Natural Language processing (NLP) modules. This is followed by a mapping phase where the attributes are detected within the English query, mapped to make the final SQL query and should then be fed into the database to retrieve the specified data and supply it to the person. The overview of our proposed work is depicted in the figure given below - Fig. 1. For now, the proposed work focuses on generation of the

same SQL query via a tongue question in English. Once the SQL query is generated the retrieval of knowledge via DB are going to be a simple task.

B. Algorithm

Creation of SQL Query via Natural Language Query

Input: Question in English textual form.

C.

Output: Query that can run in SQL

1)Tokenize the input into list of words

2)Lemmatize the list of words

3)Perform POS tagging

4) Parsed sentence = Parse using regular expressions

5)If table. attribute ∈ Parsed sentence

a) Extract them

b) Call the function SQLmap()

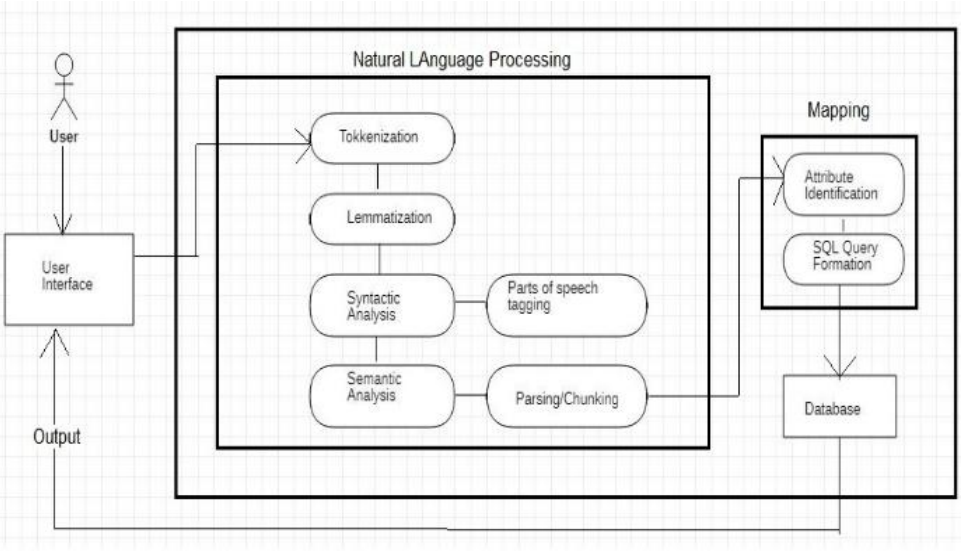

Figure 1 Proposed System Models

The system that we are presenting contains several modules which are required in order to extract only the key words and not the wasteful data that is not needed. This is often very important because presence of useless information is going to definitely reduce the general performance of the system. input file initially goes through an NLP phase followed by a mapping phase. The NLP part is made of modules such as tokenization, lemmatization, Parts Of Speech tagging (POS tagging) and parsing. The mapping part looks for the attributes that are needed and maps them into the query, thus creating an sql query.

```
Sentence after Tokenisation:
['What', 'is', 'the', 'mileage', 'of', 'Swift', 'Dzire', '?']

Tokensied words after removing punctuations:
['What', 'is', 'the', 'mileage', 'of', 'Swift', 'Dzire']

After POS tagging:
[('What', 'WP'), ('is', 'VBZ'), ('the', 'DT'), ('mileage', 'NN'), ('of', 'IN'), ('Swift', 'NNP'), ('Dzire', 'NNP')]
```

Figure 2 Tokenisation, removing punctuations and POS tagging

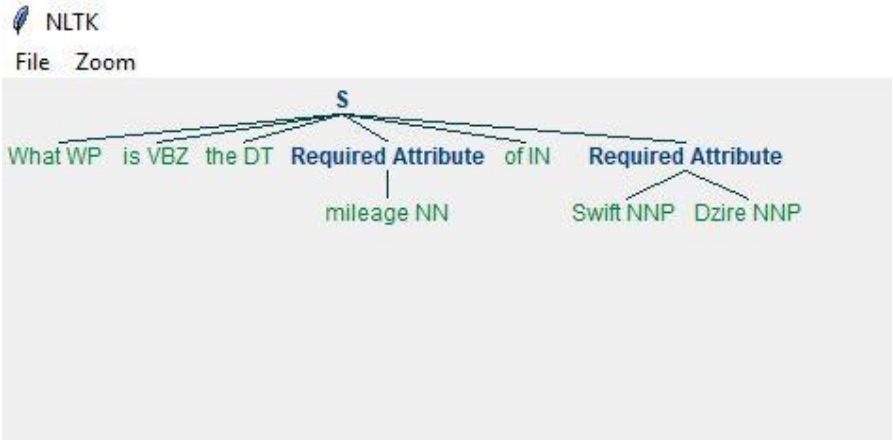

Figure 3Parse Tree

1) Tokenization: This is the first module of our system. This module breaks down the sentence into individual token or more specifically words. We apply this tokenisation as soon as we receive the input from the user and store the individual words in a list. We have made use of Tokenisation module of nltk.tokenize library in Python.

2) Lemmatization: The lemmatization module is similar to the stemming module in which the suffix or the prefix of the word is removed and the base word is given out. We have chosen to use Lemmatization instead of stemming because stemming just removes the prefix or the suffix and doesn't always give a proper word as an output, whereas stemming matches the word in the wordnet and gives a proper word as output everytime.

3)Syntactical Analysis: In this module every tokenised and lemmatized word is analysed and based on their appearance they are tagged with a Part of Speech tag. All the words are tagged and stored in a tuple and all these tuples are put in a list. We have used nlkt.pos_tag() module for this process available in the nltk library.

4) Semantic Analysis and Attribute Identification: In this module, we attempt to add all the tokens, so that the system could move forward to the formation of the sql query. An attribute is a column in the database table. We extract the attributes from the list of words that we have created and place them in the sql query accordingly. We extract the common nouns and the proper nouns from our original natural language query.

We assume that the proper noun is going to be the name of the car whose features are required by the person and the common nouns are the various features of the car that the person wants. We do so by passing the list of words through the Regular Expression mentioned below:

{<NN.?>*<NNP>?}

Figure 4 Mapping Attributes

We map these nouns into our query using the SQLmap() algorithm explained ahead

Algorithm: SQLmap()

IV. EXPERIMENTS AND RESULTS

We have taken a Vehicle Management Database for the conversion of our natural language to an sql query. We are considering a single table for which we will be creating SQL queries. The Database has columns namely ID, Make, Model, Version, Body style, price, Fuel type, Mileage. We are implementing the mentioned model on python 3 with the help of these modules - nltk, re, nltk.tag, nltk.stem, datetime, nltk tokenize, nltk corpus. The natural language queries that are given as input to the model are interrogative in nature and start with the word "What" followed by common and proper nouns. These common and proper nouns are the different attributes and constraints that need to be in the SQL query. The dataset contains NL that was created using the aforementioned format. We will now be considering a sample to see see how an input is processed through all the modules. Question: "What is the price of Ford Fiesta?". First thing that happens is the input question is split into individual words and stored in a list, i.e. tokenized. First thing that happens is the input question is split into individual words and stored in a

list, i.e. tokenized, followed by the removal of the words that are not necessary, by using stop words that is a list of words which are unimportant to the process. After this the list is lemmatized and Parts if speech tagging is done using the nltk.pos_tag() module.
After Parts of speech tagging, the given list of words is passed through a regular Expression parser which filters out the required attributes from the list of words.
This is then proceeded by mapping of the aforementioned identified attributes by using the sqlmap() Algorithm. If the list of words has no nouns that correspond to any attribute of our table we classify it as an erroneous input.

V. CONCLUSION

Even though a number of various methodologies have been used for converting natural language into an SQL query, the use of python 3 and its nltk module has proven to be the simplest and easiest and the most efficient to implement. Our work has proven good enough in presenting and explaining the exact workings of the system.

There are still some improvements that can be done to this system. These improvements are mentioned below.
Our system is still unable to create complex and nested queries that are very often required while working on a database containing multiple tables.
A speech input option can be added to this where the input is given in a spoken form and the speech is converted to text.
This system can also be integrated into a Chatbot via which user can access the database in a very interactive form.

V. REFERENCES

[1] https://www.researchgate.net/post/How__to__approach__a__system__to__convert__english__text__to__SQL
[2] Prasun Kanti Dey, Ghosh, and Subhabrata - Automatic sql query format from natural language query
[3] Garima Singh, https://www.iaees.org/publications/journals/selforganizology/articles/2016-3(3)/algorithm-to-transform-natural-language-into-SQL-queries.pdf
[4] Woods and William, The lunar sciences natural language information system
[5] https://github.com/machinalis/quepy
[6] Xiaojun Xu, Chang Liu, Dawn Song-SQLNet: generating structured queries from natural language without reinforcement learning
[7] Sathick, Javubar , Jaya - Natural language to SQL generation for semantic knowledge extraction in social web sources
[8] Bei-Bei Huang, Guigang Zhang- A natural language database interface based on a probabilistic context free grammar
[9] https://datascience.stackexchange.com/questions/31617/natural-language-to-sql-query
[10] Wan FJ. 2000. A fuzzy grammar and possibility theory – based natural language user interface for spatial queries
[11] Nandhini S, B.Viruthika, Almas Saba, Suman Sangeeta-Extracting Sql Query Using Natural Language Processing
[12] https://arxiv.org/abs/1801.06146
[13] Akshay Kulkarni, Adarsha Shivananda-Natural Language Processing Recipes
[14] https://stackoverflow.com/questions/54819075/what-are-some-of-the-ways-to-convert-nlp-to-sql
[15] https://github.com/dadashkarimi/seq2sql
[16] http://www.ling.helsinki.fi/kit/2009s/clt231/NLTK/book/ch10-AnalyzingTheMeaningOfSentences.html#querying-a-database