Natural Language to SQL: Automated Query Formation Using NLP Techniques

Sri Lalitha Y[1*], *Prashanthi* G[1], *Sravani* Puranam[1], *Sheethal Reddy* Vemula [1], *Preethi* Doulathbaji[1], *Anusha* Bellamkonda[1]

[1]Department of Information and Technology, Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad, India

Abstract. In this era of information world, given any topic, we are able to get relevant data or documents at a mouse click. The flexibility that internet provides is the user friendly language or Natural Language to search for required topic. Natural Language Querying and Retrieval has made internet popular. It is implicit for business user to understand what the business data is indicating to find better business opportunities. Querying for required data the business users are using SQL. To effectively Query such systems, the Business users has to master the Language. But many business users may not be aware of the SQL language or may not be aware of the databases and some users feel difficulty to write the long SQL Queries. Therefore, it is equally important to query the database very easily. The work here presents a case study to help the business users to type a query in Natural Language, which then converts into SQL statement and process this SQL query against the Databases and get the expected result. This work proposes QCNER approach to extract SQL properties from Natural Language Query. The proposed approach after the application of SMOTE technique depicts 92.31 accuracy over the existing models.[1]

1 Introduction

Everyone in today's society wants to be more dynamic in terms of information collection, retrieval, and sharing by utilising existing database storage systems, but they aren't familiar enough with the underlying technology. For example, consider a database, say DB. We have placed specific tables with properties that are correctly normalised within this DB database. Now, to work with the database one must be technically adept in SQL in order to run a query against the DB database. But not everyone enjoys or is capable of writing a SQL query to search a large database. So, if this problem can be solved, it would be extremely valuable. As a result, asking database queries in plain language is a highly practical and straightforward way to access data.

*Corresponding author:srilalitham.y@gmail.com

2 Literature Survey

This study suggests a way for executing database queries using a natural language interface, addressing a significant challenge in database administration for non-programmers. The article introduces a module for an interface that can translate PL/SQL instructions from natural language queries into precise results. Semantic Grammar is used in the proposed architecture to convert English statements into PL/SQL queries that may be run on a database. The system design has many phases for translating natural language queries into their PL/SQL equivalents[1].P. Parikh et al., article describes a project that intends to develop an application that translates natural language enquiries into SQL queries using a sequence-to-sequence paradigm, making SQL less difficult for course teachers in educational organisations. According to the study, this method greatly shortened the time needed for users to access the database without help from administrators. This gave educators the chance to review the data, acquire understanding of how the courses were being received and used, and make the necessary adjustments to enhance the user experience for students[2].

The difficulties of confirming the accuracy of SQL queries are covered in the essay, along with the significance of carefully crafted evaluation activities for students. The authors, Do, Quan & Agrawal, Rajeev & Rao, Dhana & Gudivada, Venkat. provide a technique for creating SQL queries that may be tailored to the intended test material by utilising RDFS to represent SQL concepts. The method is database metadata-driven and generic. To show the efficiency of the suggested strategy, an online prototype system is created[3].Chaudhari, M.S., Hire, M.A., Mandale, M.B., & Vanjari, M.S In order to save teachers' time and provide students with a wide range of fresh SQL questions, this article suggests an inverted strategy to developing SQL exercises by generating solutions before questions. The conventional approach to developing SQL exercises may be monotonous and fail to engage students' critical thinking. The suggested paradigm and its possible advantages and benefits for future SQL instruction are examined[4].M. Uma, V. Sneha, G. Sneha, J. Bhuvana and B. Bharathi project, a system that allows users to access database data via structured natural language input and SQL output is being developed. Tokenization, lemmatization, parts of speech labelling, parsing, and mapping are all steps in the process. Using regular expressions and NLP, the proposed approach translated English-language SQL queries with an accuracy of 98.89%[5].

Aditya Narhe , Chaitanya Mohite , Rushikesh Kashid , Pratik Tade, Santosh Waghmode This article examines the creation of SQL queries using natural language processing (NLP). Language analysis and information retrieval are two applications of NLP, a branch of AI. The system seeks to offer non-technical users a useful and easy method to access databases. The type of query that the user has asked may be determined by the system, which can then generate the appropriate SQL query and execute it. The significance of precise prediction in NLP-based query creation is highlighted in the paper[6].Ghosh, P.K., Dey, S., & Sengupta, S., The interaction between computers and human languages is the focus of the study of natural language processing. At the nexus of linguistics and artificial intelligence, it is an interdisciplinary research area. Its objective is to develop software that can comprehend and produce writing and voice that is similar to that of a person. Information retrieval, machine translation, and improving human-computer interaction all benefit from the use of NLP[7].

The approach for automating the conversion of natural language questions into SQL queries is presented in the article by A. Kate, S. Kamble, A. Bodkhe and M. Joshi. Although SQL is a helpful tool for managing data in a relational database management system, users who are not familiar with it may have trouble getting the information they need. The suggested approach translates natural language inquiries into SQL queries using NLP, allowing non-SQL users to access the necessary data. Additionally, for users who

prefer speaking their queries, the system offers speech-to-text conversion[8].H. Sanyal, S. Shukla and R. Agrawal. In this article the process of producing SQL queries, which are frequently difficult to construct and take a lot of time, can be made simpler by using natural language processing. SQL queries may be dynamically produced from English text inputs by using parsing, lexical analysis, synonym detection, and creation methods. With this technique, users don't need to be familiar with SQL, and queries are produced quickly and precisely. The recommended approach offers superb performance and accuracy for data production, insertion, and retrieval[9].F. Siasar djahantighi, M. Norouzifard, S. H. Davarpanah and M. H. Shenassa The use of natural language processing to communicate with databases is covered in the article. The goal is to make it possible for non-technical people to interact with the computer and receive the required data. The suggested system converts SQL queries from natural language requests using NLIDB technology[10].

3 System Design

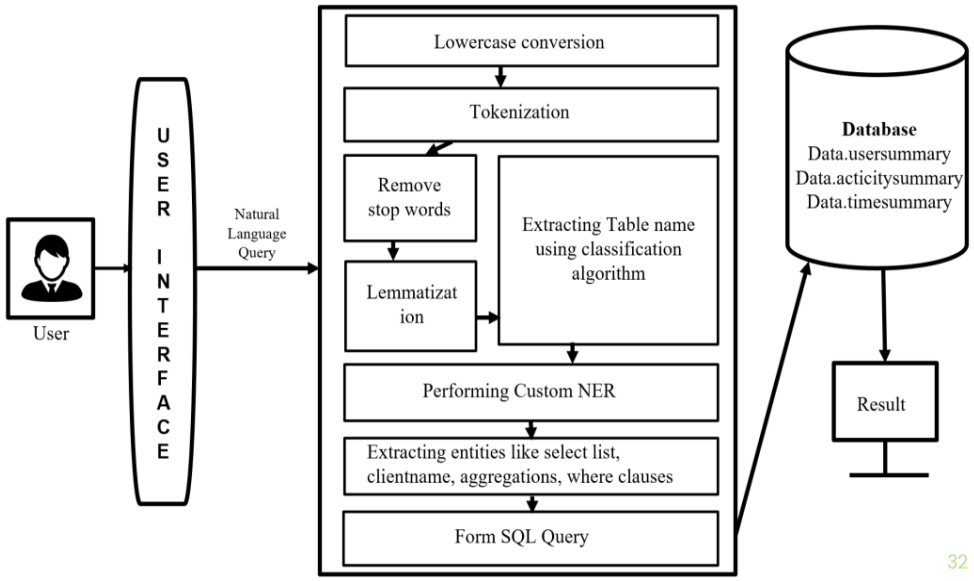

Fig. 1. Design of the system

4 Implementation

4.1 Dataset

The dataset was provided by Qualetics Company. The available data contains the data of last 12 months, consisting of several events performed in a Learning Management System (LMS). This dataset will consist of the below four parameters
1.NLP_Query
2.SQL_Query(which is for reference)
3.Reference_Table
4.data_base_connection.

SNO	NLP_Query	SQL_Query	Reference_Table	data_base_connectior
1	How many return users present in snapchat clien	select return_users from data.usersummary where date(re	data.usersummary	emqdata_qualetics
2	How many users registered for truecaller client b	select total_users from data.usersummary where date(rec	data.usersummary	emqdata_qualetics
3	How many return users are there for the remini c	select return_users from data.usersummary where date(re	data.usersummary	emqdata_qualetics
4	what is the average timespent between 7th marc	select distinct actortype from data.activitylog where client	data.timesummary	emqdata_qualetics
5	what is the average timespent between 3rd marc	select distinct actortype from data.activitylog where client	data.timesummary	emqdata_qualetics
6	How many users registered previous week for yo	select new_users from data.usersummary where date(rec	data.usersummary	emqdata_qualetics
7	How many active users present in voot client bet	select active_users from data.usersummary where date(re	data.usersummary	emqdata_qualetics
8	How many total users registered in last quarter f	select new_users from data.usersummary where date(rec	data.usersummary	emqdata_qualetics
9	How many return users present in pinterest clien	select return_users from data.usersummary where date(re	data.usersummary	emqdata_qualetics
10	what are the different actortype of Metamask cl	select distinct actortype from data.activitylog where client	data.timesummary	emqdata_qualetics
11	what are the distinct actor registered between 1	select distinct actortype from data.activitylog where client	data.activitysummary	emqdata_qualetics
12	what is the count of inactive users present in goc	select active_users from data.usersummary where date(re	data.usersummary	emqdata_qualetics
13	what is the sessionid of snapspeed client?	select distinct actortype from data.activitylog where client	data.timesummary	emqdata_qualetics
14	How many active users present in myntra client f	select active_users from data.usersummary where date(re	data.usersummary	emqdata_qualetics

Fig. 2. Dataset records

The databases provided by the company contains more than 70000 records which specifies about usage of product of the client. The databases has above 30 clients.

Sample Databases

data.usersummary: This particular table tracks the total number of users in the system along with their status like active , inactive etc.

Data Output Explain Messages Notifications

	id integer	clientid bigint	clientname text	newusers double precision	total_users double precision	active_users double precision	inactive_users double precision	return_users double precision	created timestamp without time zone
1	4382	72	telegram	1	11	1	10	0	2022-01-04 08:16:56.132132
2	4400	72	telegram	0	11	1	10	1	2022-01-06 08:16:56.005582
3	4409	72	telegram	0	11	2	9	2	2022-01-07 08:16:55.527058
4	4420	72	telegram	0	11	1	10	1	2022-01-08 08:16:56.127864
5	4389	90	snapchat	1	24	1	23	0	2022-01-05 08:16:54.412757

Fig. 3. data.usersummary

data.activitysummary: This particular table tracks overall activity performed by users along with their respective attributes.

Data Output Explain Messages Notifications

	id integer	clientname text	actionname character varying (150)	actor character varying (150)	actortype character varying (50)	contextname character varying	contexttype character varying (50)	sessionid character varyin
1	2411271	yono SBI	Start Survey	5		Survey		9c64d95f-fa03-4
2	2411272	yono SBI	Course Start	5		Course		9c64d95f-fa03-4
3	2411273	yono SBI	Finish/Exit Course	5		Course		9c64d95f-fa03-4
4	2411274	yono SBI	Submit Response	5		Survey Response		9c64d95f-fa03-4
5	2411275	yono SBI	View Question	5		Survey Question		9c64d95f-fa03-4
6	2411277	yono SBI	PageView	5		Page		9d053bb6-aad5-
7	2411278	yono SBI	PageView	5		Page		9d053bb6-aad5-
8	2411279	yono SBI	PageView	5		Page		9d053bb6-aad5-
9	2411280	yono SBI	PageView	5		Page		9c64d95f-fa03-4
10	2411281	yono SBI	PageView	5		Page		9c64d95f-fa03-4

Fig. 4. data.activitysummary

data.timesummary: This particular table tracks the time dimension (time taken) for different actions under different contexts.

Data Output Explain Messages Notifications

	id integer	clientname text	actor character varying (150)	actortype character varying (50)	contextname character varying	contexttype character varying (50)	sessionid character varying (50)	recordtimesta timestamp wi
1	4363663	mcdonalds	8	System	Page	Page	600291d1-81e3-4f5b-9556-c683e59dbc70	2022-04-01 0(
2	4363664	mcdonalds	8	System	Page	Page	600291d1-81e3-4f5b-9556-c683e59dbc70	2022-04-01 0(
3	4363665	mcdonalds	8	System	Page	Page	600291d1-81e3-4f5b-9556-c683e59dbc70	2022-04-01 0(
4	4363666	mcdonalds	8	System	Page	Page	600291d1-81e3-4f5b-9556-c683e59dbc70	2022-04-01 0(
5	4363667	mcdonalds	8	System	Page	Page	600291d1-81e3-4f5b-9556-c683e59dbc70	2022-04-01 0(
6	4363668	mcdonalds	8	System	Page	Page	600291d1-81e3-4f5b-9556-c683e59dbc70	2022-04-01 0(
7	4363669	mcdonalds	8	System	Page	Page	600291d1-81e3-4f5b-9556-c683e59dbc70	2022-04-01 0(
8	4363670	mcdonalds	8	System	Page	Page	600291d1-81e3-4f5b-9556-c683e59dbc70	2022-04-01 0(
9	4363671	mcdonalds	8	System	Page	Page	600291d1-81e3-4f5b-9556-c683e59dbc70	2022-04-01 0(
10	4363672	mcdonalds	8	System	Page	Page	600291d1-81e3-4f5b-9556-c683e59dbc70	2022-04-01 0(

Fig. 5. data.timesummary

We are really grateful for the company for sponsoring us the dataset and the databases which helped us a lot to move ahead in this work.

4.2 Algorithm

4.2.1 Smote

For the minority class, SMOTE uses oversampling to produce phoney samples. This approach helps address the overfitting problem caused by random oversampling. It concentrates on the feature space to produce new examples by interpolating between positive cases that are close to one another.

4.2.2 Random Forest

Random Forest is an ensemble learning method that combines many decision trees to improve accuracy and decrease overfitting. The classification in a Random Forest is determined by a majority vote among the decision trees, and each tree is trained using a randomly chosen portion of the training data. Random Forest is well-known for its high accuracy and resistance to noise and outliers in the data.

4.2.3 Naive Bayes

A Naive Bayes classifier,is used to carry out classification tasks. The classifier's foundation is the Bayes theorem. When dealing with document classification problems, such as determining whether or not a document belongs to a particular table category, multinomial Nave Bayes is widely utilised. One of the characteristics or predictors the classifier uses is the frequency of the terms in the document.

4.2.4 SVM

A potent technique used for classification and regression applications is called the Support Vector Machine (SVM). A hyperplane that best divides the classes in the data is found via SVM. SVM can handle nonlinear data by translating it into a higher-dimensional space where classes may be distinguished by a linear hyperplane. SVM is renowned for its excellent accuracy and capacity to handle complicated datasets, but the choice of the kernel function and hyperparameters might affect its performance.

4.2.5 KNN (K Neighrest Negihbours)

A non-parametric technique called K-Nearest Neighbour (KNN) is utilised for both classification and regression applications. Finding the k nearest data points in the training set and assigning the most prevalent class among them allows KNN to classify a new data point. KNN is straightforward to use and frequently works well on tiny datasets.

4.2.6 NER (Named Entity Recognition)

The recognition and classification of named entities is the focus of the named entity recognition process in NLP. Using the raw and structured text, the specified entities are divided into people, organisations, locations, money, time, and so forth. In essence, named entities are identified and sorted into several groupings.NER systems are developed using a variety of linguistic strategies as well as statistical and machine learning methodologies. NER offers a variety of uses for business or project-related purposes. Before classifying an object into the best suitable class, the NER model first detects it. Here are a few instances of named entities:

1. Organization
2. Person
3. Location
4. Date etc

Fig. 6. Entities Recognition From Text

The disadvantage of NER is that spacy already has a pipeline for named recognition. For instance, Flipkart may be listed as a person rather than an ORG, and other necessary entities may not be listed. Consequently, a special NER model is required.

In order to extract domain-specific entities from unstructured text, such as contracts or financial documents, users of custom NER can create their own AI models. Before making a Custom NER project accessible for consumption, developers can repeatedly label data, train, assess, and improve model performance.

4.2.7 Workflow

In this paper, we proposed a system that converts SQL (Structured Query Language), a programming language for databases, into plain language queries. The stages for converting a query from natural language to a database query (SQL) are as follows, and they will be carried out in order.

1. The first step in the procedure is to tokenize, lemmatize, and remove stop words from the natural language question.
2. The target column is then factorised, and the Multinomial Naive Bayes method is used to analyse the pre-processed data.Balancing methods are used when the dataset is unbalanced.
3. A customised NER model is then created to extract various entities from the query, and the entities and associated table names are stored in a dictionary.
4. Using the pypika library and this data, a SQL query is then created. After using the psycopg2 library to perform the SQL query, the user receives the expected results.

5 Evaluation

Table 1. Comparison of Different Algorithms

Random Forest Classifier Algorithm	Accuracy for Table Classification is : 92.31 precision recall f1-score support 0 0.97 0.94 0.96 36 1 0.94 0.85 0.89 34 2 0.87 0.97 0.92 34 accuracy 0.92 104 macro avg 0.93 0.92 0.92 104 weighted avg 0.93 0.92 0.92 104
Support Vector Classifier Algorithm	Accuracy for Table Classification is : 89.42 precision recall f1-score support 0 0.94 0.92 0.93 36 1 0.83 0.85 0.84 34 2 0.91 0.91 0.91 34 accuracy 0.89 104 macro avg 0.89 0.89 0.89 104 weighted avg 0.90 0.89 0.89 104
Multinomial Naïve Bayes	Accuracy for Table Classification is : 87.5 precision recall f1-score support 0 0.95 0.97 0.96 36 1 0.87 0.76 0.81 34 2 0.81 0.88 0.85 34 accuracy 0.88 104 macro avg 0.87 0.87 0.87 104 weighted avg 0.88 0.88 0.87 104
KNeighbors Classifier Algorithm	Accuracy for Table Classification is : 83.65 precision recall f1-score support 0 0.93 0.72 0.81 36 1 0.75 0.88 0.81 34 2 0.86 0.91 0.89 34 accuracy 0.84 104 macro avg 0.85 0.84 0.84 104 weighted avg 0.85 0.84 0.84 104

6 Experimental Results

```
Enter the Query : what is the id of gopro client?
Query is : what is the id of gopro client?
Accuracy for Table Classification is :  83.65
Extracted Table Name is :  data.activitysummary

Entities Are :  {'SelectList': 'id', 'clientname': 'gopro'}
SQL Query is :  SELECT id FROM data.activitysummary WHERE clientname='gopro'
Ans =  2505722
Ans =  2505723
```

Fig. 7. Output of data.activitysummary table

```
Enter the Query : How many users registered for snapchat client between 23rd april 2021 and 17th jan 2022?
Query is : How many users registered for snapchat client between 23rd april 2021 and 17th jan 2022?
Accuracy for Table Classification is :  83.65
Extracted Table Name is :  data.usersummary

Entities Are :  {'Aggregation': 'many', 'SelectList': 'inactive_users', 'clientname': 'snapchat', 'recordtimestamp': '2021-04
-23', 'recordtimestamp1': '2022-01-17'}
SQL Query is :  SELECT COUNT(inactive_users) FROM data.usersummary WHERE clientname='snapchat' AND recordtimestamp>='2021-04-
23' AND recordtimestamp<='2022-01-17'
Ans =  6
```

Fig. 8. Output of data.usersummary table

```
Enter the Query : what is the average timespent by msword client?
Query is : what is the average timespent by msword client?
Accuracy for Table Classification is :  83.65
Extracted Table Name is :  data.timesummary

Entities Are :  {'Aggregation': 'average', 'SelectList': 'timespent', 'clientname': 'msword'}
SQL Query is :  SELECT AVG(timespent) FROM data.timesummary WHERE clientname='msword'
Ans =  8.746869761979347
```

Fig. 9. Output of data.timesummary table

7 Conclusion and Future Enhancements

It takes time and effort to create a series of queries for learning SQL, which might increase the workload for a teacher or instructor. The method we used for building SQL exercises is preferable to the traditional one. The suggested approach performs better for straightforward queries but not for complicated ones. Future work should focus on creating a model that can be used for complicated queries that contain joins, constraints, group by clauses, having clauses, etc.

References

1. Swapnil Kanhe, Pramod Bodke, Akshay Chikhale, Vaibhav Udawant, "SQL Generation and PL/SQL Execution from Natural Language Processing", INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume **04**, Issue 02, (February 2015).
2. P. Parikh et al., "Auto-Query - A simple natural language to SQL query generator for an e-learning platform," IEEE Global Engineering Education Conference (EDUCON), Tunis, Tunisia, 2022, pp. 936-940, doi: 10.1109/EDUCON52537.2022.9766617, (2022).
3. Do, Quan & Agrawal, Rajeev & Rao, Dhana & Gudivada, Venkat. "Automatic Generation of SQL Queries". ASEE Annual Conference and Exposition, Conference Proceedings. 10.18260/1-2—20112, (2014).
4. Chaudhari, M.S., Hire, M.A., Mandale, M.B., & Vanjari, M.S. "Structural Query Language Question Creation by using Inverse Way", (2021).
5. 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, doi: 10.1109/ICCIDS.2019.8862080, (2019).
6. Aditya Narhe , Chaitanya Mohite , Rushikesh Kashid , Pratik Tade, Santosh Waghmode, "SQL Query Formation for Database System using NLP", INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume **08**, Issue 12, (2019).
7. Ghosh, P.K., Dey, S., & Sengupta, S. "Automatic SQL Query Formation from Natural Language Query", (2014).
8. 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, India,pp. 488-491, doi: 10.1109/ICECA.2018.8474639, (2018).
9. H. Sanyal, S. Shukla and R. Agrawal, "Natural Language Processing Technique for Generation of SQL Queries Dynamically,"6th International Conference for Convergence in Technology (I2CT), Maharashtra, India, pp. 1-6, doi: 10.1109/I2CT51068.2021.9418091, (2021).
10. F. Siasar djahantighi, M. Norouzifard, S. H. Davarpanah and M. H. Shenassa, "Using natural language processing in order to create SQL queries," International Conference on Computer and Communication Engineering, Kuala Lumpur, Malaysia, pp. 600-604, doi: 10.1109/ICCCE.2008.4580674, (2008).