



Bilingual Text-to-SQL Report Generation for WooCommerce

Dr. Naveenkumar T Rudrappa Ayman Shaikh Neha Mule Prajwal Naik Raksha Hebbar
Department of Computer Science and Engineering *Department of Computer Science and Engineering* *Department of Computer Science and Engineering* *Department of Computer Science and Engineering* *Department of Computer Science and Engineering*
KLS VEDIT, Haliyal KLS VEDIT, Haliyal KLS VEDIT, Haliyal KLS VEDIT, Haliyal KLS VEDIT, Haliyal
INDIA INDIA INDIA INDIA INDIA

Abstract: This paper presents an approach which enables seamless conversion of Natural language Query input queries into SQL Query efficiently for WooCommerce users. Common business owners who are the prime users of WooCommerce are usually not technically sound to query with their database in SQL. Therefore, we have introduced our system “Text-to-SQL report generation for WooCommerce” bridging the gap between non-technical users and Database interaction. Our system also provides bilingual support as an additional functionality for the WooCommerce users which is not provided in the existing systems. This system enables users to Input queries in both English and Hindi through text or voice. The system then generates the output in the Tabular data and Visual Graph formats facilitating better analysis and decision-making. By eliminating the need of SQL expertise, WooCommerce users can simply use our system which offers a user-friendly database interaction with bilingual support. This undertaken project represents a significant advancement towards empowering WooCommerce users to Interact with their Database effectively.

I. INTRODUCTION

A visionary language known as Structured Query Language (SQL) which emerged in the 1970's was specifically designed for managing relational database. Its syntax was designed to handle complex queries which made it essential in database management system. SQL's power lies in its ability to interact effortlessly with the database, allowing users to retrieve, update and manipulate data easily. SQL plays an integral role in enhancing human-computer interaction. For users without a technical background, Using SQL poses a technical barrier. The difficult and confusing syntax with its structured commands often confuses those who are not well-versed in programming.

As technology advanced in the realms of artificial intelligence and natural language processing (NLP), a significant shift was noticed with the way humans interacted with machines in human-like language. This recognition led to the realization for building new potential systems for converting natural language queries into SQL queries. This would allow individuals with diverse backgrounds including those without technical background to communicate with their databases.

This system is specifically designed for the users of WooCommerce who are not technically proficient. These people face challenges while interacting and retrieving their business analytics from WooCommerce as they lack SQL expertise. So, to help such users we have designed our novel system that bridges the gap between non-technical users and database interaction enabling seamless Conversion of plain language queries into SQL queries and draw Graph visualizations for better decision-making. These visualizations will help the users to understand and make informed decisions related to their business.

Additionally, our system provides bi-lingual support where the users can give input in both English and Hindi language through Text/Voice to retrieve the output in Tabular and graphical forms. This bilingual support empowers users from diverse backgrounds to use WooCommerce as their online business platform which will consequently attract more business to WooCommerce.

II.LITERATURE REVIEW

Xu Yiqiu, Wang Liwei, and Yan Shi proposed a system for "Domain Specific Query Generation from Natural Language Text" outlines an automated method for generation SQL from natural language, focusing on software requirements specifications. It addresses challenges such as disclosure and semantic intricacies, formal representation of knowledge within a domain and semantic disambiguation. Evaluation metrics shot a promising result with the average between 84.2% and recall of 75.4% [1]

G. Koutrika, A. Simitsis and Y. E. Ioannidis proposed a system for "A Simple Guide to Implement Data Retrieval Through Natural Language Database Query Interface (NLDQ)" presents a pioneering NLP system enabling English-based database querying with SQL expertise. It purposes a literal-based approach SQL. It proposes a literal-based approach to translate natural language queries into SQL for execution. The methodology involves tokenization, parsing, and SQL generation stages. [2]

T. Mahmud, K. M. Azharul Hasan, M. Ahmed and Thwoi Hla Ching Chak proposed a system for "Natural Language Interface to Database (NLIDB) for Query with Temporal Aspect" focuses on enhancing NLIDB by addressing challenges in handling queries and implementing unit conversion. It builds upon an existing NLIDB system, introducing modifications to handle various question types and facilitate unit conversion, improving NLIDB capabilities significantly. [3]

A. Iftikhar, E. Iftikhar and M. K. Mehmood Proposed a system for "TiQi: A Natural Language Interface for Querying Software Data" introduces TiQi, a web-based interface for querying extensive software data using natural language. TiQi visualizes data, translates queries into SQL, and executes them. Evaluation results show promising correctness rates, with ongoing enhancements planned for future versions. [4]

J. Lin et al Proposed a system for "Natural Language Processing Technique for Generation of SQL Queries Dynamically" explores Natural language processing methodologies and parsing techniques integrated with Structured query language for database interactions. It emphasizes NLP importance in human-machine communication and introduces a comprehensive system for dynamically generating and optimizing SQL queries, aiming for high-performance results. [5]

E. U. Reshma and P. C. Remya proposed a system "A Rule-Based Approach for NLP-Based Query Processing" addresses the challenge for developing NLIDBs tailored for non-experts. It proposes a rule-based system using Context-Free Grammar (CFG) to translate natural language queries [2] into SQL, providing a user-friendly interface for database interactions without requiring SQL expertise [6]

Solanki, Arun & Kumar, Ashutosh proposed a system "The Study on Natural Language Interface of Relational Databases" introduces a WordNet-based NLIDBs framework to improve natural language querying [2] accuracy and reliability. It leverages WordNet, Ontology, and DRS to translate queries into SQL seamlessly, enhancing search accuracy. The system's modules, tested with SQL Server database, show efficiency in single-table queries. [7]

A. Kate, S. Kamble, A. Bodkhe and M. Joshi proposed a system "A Review of Different Approaches in Natural Language Interface to Databases" provides insights into NLIDB systems, which facilitate user-database interactions using natural language, Eliminating the need for SQL expertise. Various NLIDB architectures and components are examined, highlighting the challenges and ongoing research in query optimization algorithms for broader NLIDB adoption. [8]

T. Ahmad and N. Ahmad proposed a system "Conversion of Natural Language Query [2] to SQL Query" presents a solution to simplify NLQ to SQL conversion, targeting users unfamiliar with SQL, particularly Training and Placement cell officers. The proposed NLP model accommodates both text and speech inputs,

integrating speech-to-text conversion for seamless query execution. The system aims to enhance database access for non-SQL-proficient users, offering a user-friendly interface with speech recognition capabilities for effortless query transformation. [9]

The project develops an advanced system with a three-tier architecture to facilitate communication between non-expert users and databases. Using production rules and a predefined data dictionary, the system processes user input to generate SQL queries, demonstrating enhanced recall, accuracy, and precision compared to existing systems. Published in BVICAM's International Journal of Information Technology, the study contributes to various fields, emphasizing the importance of intelligent interfaces to improve database communication efficiency. [10]

M. Uma, V. Sneha, G. Sneha, J. Bhuvana and B. Bharathi paper presents Data Agnostic RoBERTa-based Natural Language to SQL Query a novel NL2SQL approach enabling natural language interaction with databases without SQL, prioritizing data privacy through table schema reliance. Utilizing RoBERTa embeddings and LSTM sub models, it achieves 76.7% accuracy, showcasing zero-shot learning capability. Contributions include a data-agnostic model and innovative use of RoBERTa for embeddings, addressing usability and privacy concerns. [11]

T. -H. -Y. Vuong, T. -T. -T. Nguyen, N. -T. Tran, L. -M. Nguyen and X. -H. Phan proposed a system "Formation of SQL from Natural Language Query using NLP" addresses users' challenges with database languages by proposing an NLP system for interacting with a railways reservation database through natural language questions converted into SQL queries. Achieving 98.89% accuracy, it simplifies data extraction, crucial for non-technical users seeking information from databases. [12]

B. Xu et al proposed a system "Human Language Question to SQL Query Using Deep Learning" discusses challenges in translating human language into structured queries, essential for database interaction. It evaluates existing approaches leveraging syntax parsing and deep learning techniques. Despite advancements, automatic translation hurdles persist, emphasizing the need for ongoing improvements in query relevance and quality. [14]

D. Pal, H. Sharma and K. Chaudhuri proposed a system Natural Language Database Querying system addressing SQL complexity, integrating deep learning and traditional parsing techniques. NADAQ enhances query relevance, rejecting irrelevant queries and suggesting candidates in natural language. With 90% accuracy in real-world systems, it provides a powerful querying interface approaching commercial standards, benefiting users unfamiliar with SQL. [15]

H. Sanyal, S. Shukla and R. Agrawal proposes a graph-based solution for explaining structured queries in natural language, addressing the challenge of providing textual explanations for form-based query interfaces. Different graph traversal strategies are presented for efficient exploration, facilitating the composition of query descriptions. Promising results demonstrate the approach's potential and pave the way for adaptive translation methods and query association mining for further research. [16]

III. PROPOSED SYSTEM

First, the user gives the input to the system through the web interface using Text or Voice in either English or Hindi Language. Upon submission, the input goes through Text validation. Valid inputs are then forwarded to the Gemini model which is our core component for interpreting the user inputs. Then these plain language queries are translated into SQL queries. Based on these queries the Data is retrieved from the database and the output is returned in the tabular form or graphically represented. Now these are the steps of implementation:

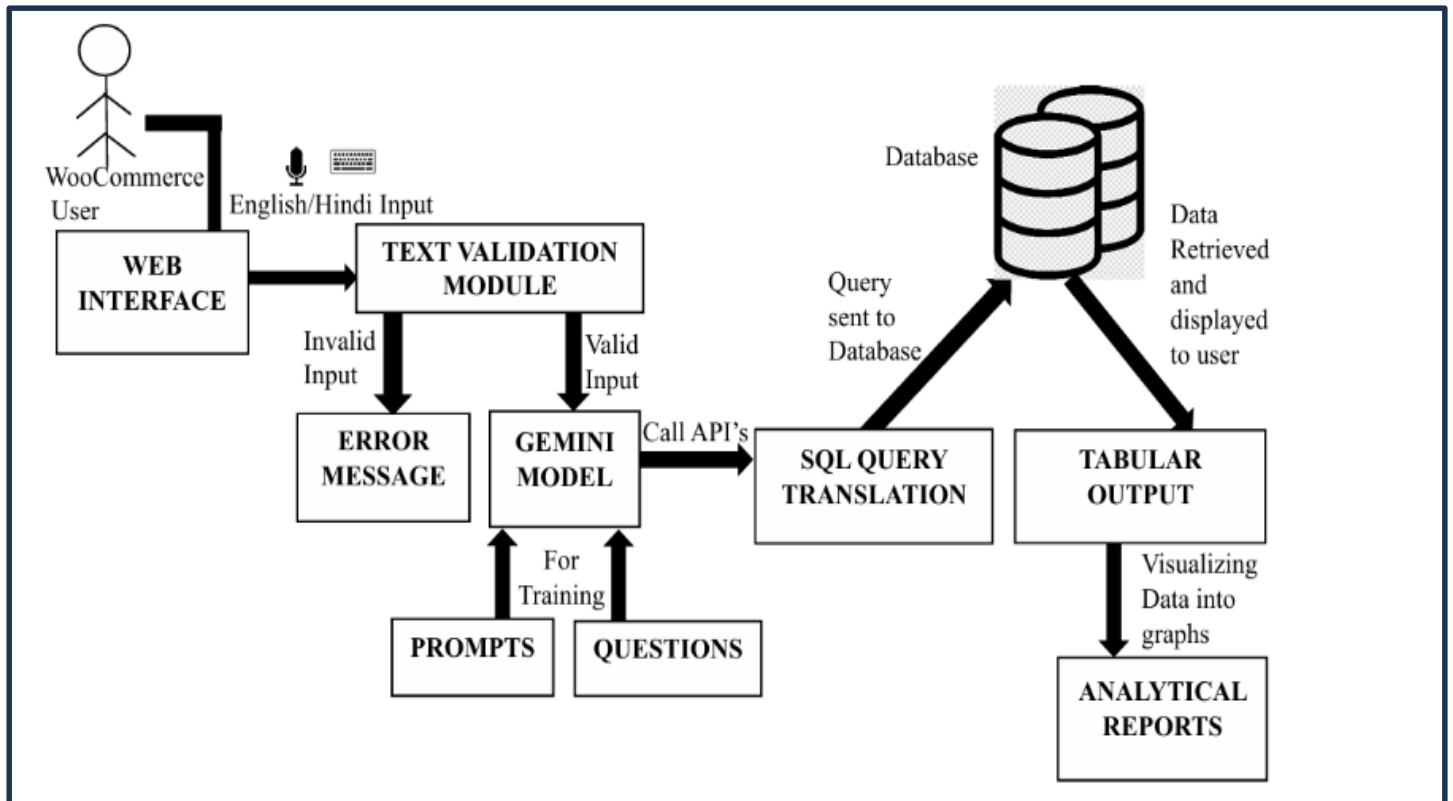


Figure 1: Intelligent Bilingual Query Processing Architecture

Step 1: Query is entered by the user in English or Voice Search in Hindi via Streamlit web interface. For example, Give the products with unit price more than 20.

Step 2: Gemini refines and interprets the user input leveraging its trained algorithms and knowledge of prompts and questions to understand the user query.

Step 3: Then the refined queries are translated into SQL queries through the dedicated Gemini API's.

Step 4: The data is then retrieved from the database by executing the SQL queries.

Step 5: The fetched Data from the database is then Visualized using the Matplotlib python module.

Step 6: The visualized Data can then be downloaded into a pdf file for Report Analysis by the user.

IV.IMPLEMENTATION

We have considered a simple Products table with attributes -ProductID, ProductName, Description, QuantityInStock, UnitPrice. We have Implemented our system in Python3 and using the following packages: PyMySQL, pandas, matplotlib, NumPy, speech recognition, google trans, Streamlit etc. The type of queries that are given to the system as input are:

1. Show me the quantity stock of the products.
2. Products with unit price more than 20.
3. Show me the products with unit price in between 20 to 50.

We can also give queries in Hindi language as voice input or Text. Some example queries are:

4. उत्पादों को उनके विवरण के साथ सूचीबद्ध करें |
5. मुझे टीशर्ट का स्टॉक की मात्रा दिखाओ |

Let us now consider 2 examples to illustrate the working of our system on the basis of English input and Hindi Input :

1.The input query is: “Show products with its unit price.” The English input query is processed by the Gemini model and translated into SQL query. The translated query is “SELECT product_name , unit_price FROM products;”.

2.The Hindi voice input query is” मुझे टीशर्ट का स्टॉक की मात्रा दिखाओ |” The voice is first translated into english by googletrans and then the english text is sent to Gemini for translation into SQL query. The speech_recognition module is used in this case.

After the query is translated , the SQL query is processed and data is retrieved from the database. The data is presented in a tabular format and then Visualized in the form of bar graphs or line graphs for better decision-making by the users. This data is formed into a pdf report using ReportLab and it can be downloaded and stored locally by the users.

V.RESULTS

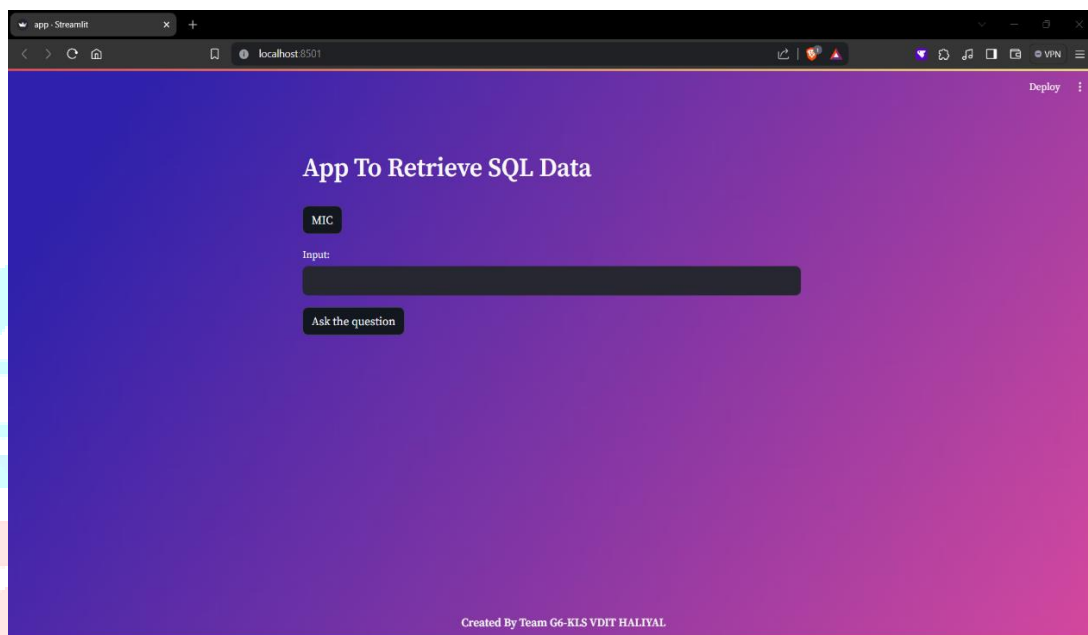


Figure 2: Web Interface for User-Database Interaction

Users Input queries through text or voice via a microphone button, then execute them with a single click. By accommodating bilingual support, the system enhances accessibility and efficiency.

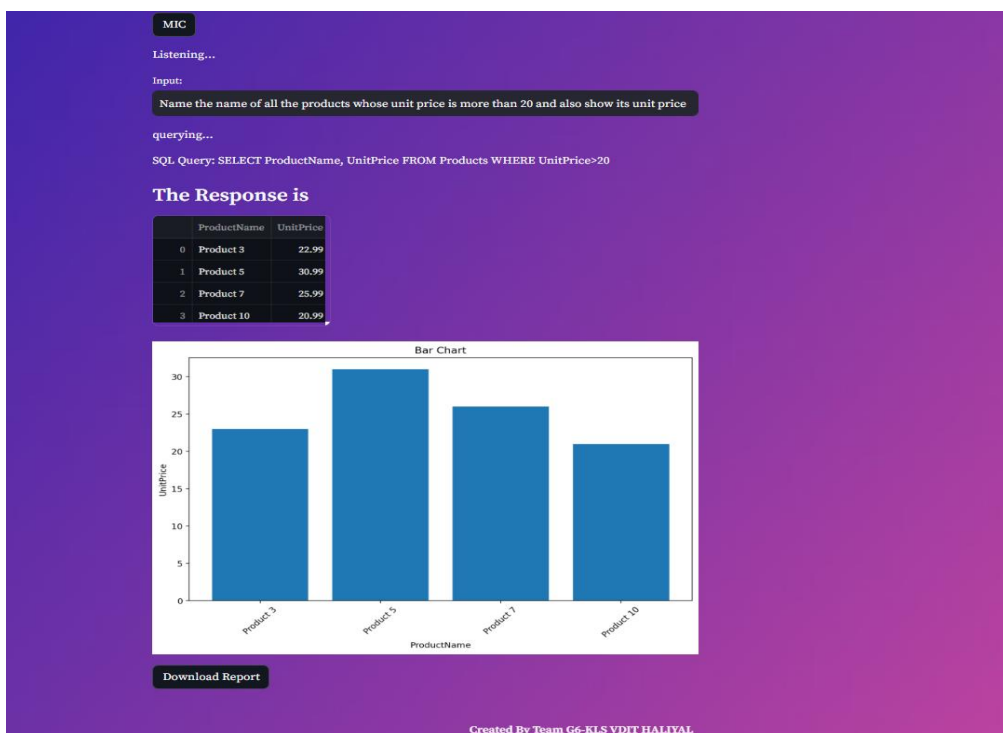


Figure 3: Voice input query and generating output

The users can query using voice commands or through text and the response is generated by the system. The output is returned in the tabular form and its corresponding bar graph is generated to visualize the data. This feature allows ease of understanding data for the WooCommerce user

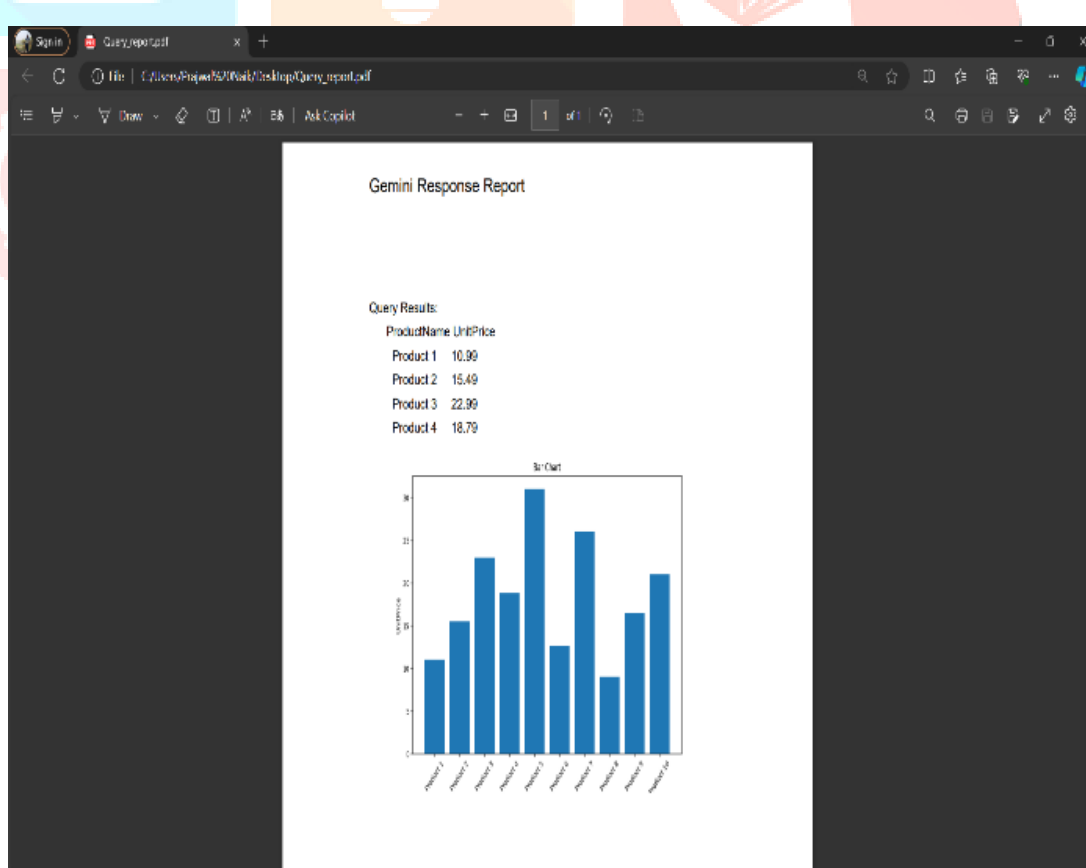


Figure 4: Analytics report Downloaded in PDF format

Once the data is retrieved and visualized in the form of graphs, the final analytics report can be downloaded as PDF by the WooCommerce user as illustrated in Figure 4.

CONCLUSION

Our system simplifies database interaction through natural language queries and speech translation capabilities. It empowers WooCommerce users to extract valuable insights from their database effortlessly without the prior knowledge of SQL. Additionally, our platform facilitates Analytics Report Generation to make informed decisions and derive meaningful insights from their data

REFERENCES

- [1] Xu Yiqiu, Wang Liwei and Yan Shi, "The study on natural language interface of relational databases," 2010 The 2nd Conference on Environmental Science and Information Application Technology, Wuhan, 2010
- [2] G. Koutrika, A. Simitsis and Y. E. Ioannidis, "Explaining structured queries in natural language," 2010 IEEE 26th International Conference on Data Engineering (ICDE 2010), Long Beach, CA, USA, 2010.
- [3] T. Mahmud, K. M. Azharul Hasan, M. Ahmed and Thwoi Hla Ching Chak, "A rule based approach for NLP based query processing," 2015 2nd International Conference on Electrical Information and Communication Technologies (EICT), Khulna, Bangladesh, 2015.
- [4] A. Iftikhar, E. Iftikhar and M. K. Mehmood, "Domain specific query generation from natural language text," 2016 Sixth International Conference on Innovative Computing Technology (INTECH), Dublin, Ireland, 2016.
- [5] J. Lin et al., "TiQi: A natural language interface for querying software project data," 2017 32nd IEEE/ACM International Conference on Automated Software Engineering (ASE), Urbana, IL, USA, 2017.
- [6] E. U. Reshma and P. C. Remya, "A review of different approaches in natural language interfaces to databases," 2017 International Conference on Intelligent Sustainable Systems (ICISS), Palladam, India, 2017.
- [7] Solanki, Arun & Kumar, Ashutosh. (2018). A system to transform natural language queries into SQL queries. International Journal of Information Technology.
- [8] A. Kate, S. Kamble, A. Bodkhe and M. Joshi, "Conversion of Natural Language Query to SQL Query," 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2018.
- [9] T. Ahmad and N. Ahmad, "A Simple Guide to Implement Data Retrieval through Natural Language Database Query Interface (NLDQ)," 2019 8th International Conference System Modeling and Advancement in Research Trends (SMART), Moradabad, India, 2019.
- [10] D. A. Poetra, T. Esterina Widagdo and F. N. Azizah, "Natural Language Interface to Database (NLIDB) for Query with Temporal Aspect," 2019 International Conference on Data and Software Engineering (ICoDSE), Pontianak, Indonesia, 2019.
- [11] M. Uma, V. Sneha, G. Sneha, J. Bhuvana and B. Bharathi, "Formation of SQL from Natural Language Query using NLP," 2019 International Conference on Computational Intelligence in Data Science (ICCIDS), Chennai, India, 2019.
- [12] T. -H. -Y. Vuong, T. -T. -T. Nguyen, N. -T. Tran, L. -M. Nguyen and X. -H. Phan, "Learning to Transform Vietnamese Natural Language Queries into SQL Commands," 2019 11th International Conference on Knowledge and Systems Engineering (KSE), Da Nang, Vietnam, 2019.

- [13] K. Ahkoul and M. Machkour, "Human Language Question To SQL Query Using Deep Learning," 2019 Third International Conference on Intelligent Computing in Data Sciences(ICDS),Marrakech,Morocco, 2019.
- [14] B. Xu et al., "NADAQ: Natural Language Database Querying Based on Deep Learning," in IEEE Access, vol. 7, 2019.
- [15] D. Pal, H. Sharma and K. Chaudhuri, "Data Agnostic RoBERTa-based Natural Language to SQL Query Generation," 2021 6th International Conference for Convergence in Technology(I2CT),Maharashtra,2021
- [16] H. Sanyal, S. Shukla and R. Agrawal, "Natural Language Processing Technique for Generation of SQL Queries Dynamically," 2021 6th International Conference for Convergence in Technology (I2CT), Maharashtra, India, 2021.

