International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

VoxDB: Voice Command Based Data Retrieval System

Gowri M Nair[*1], Shabna Shajahan[*2], Clair Mary Mathew[*3], Maria Joseph[*4], Rakhi Ramachandran Nair[*5]
College of Engineering Kidangoor, Kerala, India
[1]gowrimnair19@gmail.com, [2]shabnashajahan003@gmail.com, [3]clairmarymathew@gmail.com,
[4]mariajoseph2310@gmail.com, [5]rakhinairtvla@gmail.com

***Abstract*—Voice-based database interaction has gained prominence as an intuitive and efficient method for retrieving information. This paper presents a literature survey on speech-to-SQL query generation, emphasis on a rule-based approach implemented in our project. While machine-learning-based techniques , they often require extensive training data and may struggle with adaptability. Rule-based approach can ensures greater accuracy and consistency by explicitly mapping natural language components to structured SQL queries. For real-time speech-to-text conversion Our system integrates webkitSpeechRecognition API, followed by Natural Language Processing (NLP) techniques for query generation. The system processes the input text by manually removing stopwords using a predefined list of stopwords. The extracted entities and conditions from the input are then structured into SQL queries based on predefined database schema rules. This survey examines existing research in speech recognition, NLP-based text processing, and rule-based query generation, highlighting their advantages and limitations. Our findings suggest that especially for domain-specific databases a hybrid approach combining rule-based methods with NLP techniques significantly improves SQL query accuracy. The structured framework enhances interpretability, adaptability, and efficiency, making it a reliable alternative to purely machine-learning-driven models.This paper emphasizing the effectiveness of a rule-based approach in ensuring precise and structured SQL query formation also provides insights into the methodologies, challenges, and advancements in speech-driven database retrieval.**

***Index Terms*—NLP, Speech Recognition, SQL Query Generation**

I. Introduction

With the increasing reliance on digital databases, enabling users to interact with data through natural language has become a significant area of research. Non-technical users have difficulty in using traditional database management systems require users to have knowledge of Structured Query Language (SQL). To address this challenge, allowing users to retrieve information through spoken commands,voice-based database interaction has emerged as a practical solution,. This allow users to access data without the need for direct SQL query writing, making data access more user-friendly and efficient. Speech-to-SQL query generation involves processing natural language input and converting it into a structured SQL query that can be executed on a database. Rule-based approaches being widely used due to their accuracy and deterministic nature.Other approaches that rely on complex models, a rule-based method follows predefined rules to extract relevant terms from user input and generate structured queries. Rule-based systems that can be easily customized to fit predefined database schemas and user requirements and are particularly effective for domain-specific applications. And, such systems ensure a transparent and explainable query formation process, making them more interpretable for end users. In order to ensure accuracy and consistency in query formation,our project implements a rule-based approach for SQL query generation.The real-time speech-to-text conversion is acheived via webkitSpeechRecognition API , followed by manual stopword removal to refine the extracted text. Instead of relying on advanced NLP models, our system processes user input using predefined rules, it finds the key terms and conditions from the extracted text and use them to convert into to structured SQL queries. This method ensures structured query generation without requiring extensive computational resources.

Speech recognition is the first step in the system's structured workflow,then the next step is text preprocessing, after that we move on to query generation,and finally the execution of the query generated and retrieve data. User input is processed to remove stopwords words,maping words into the root word, extract relevant keywords and conditions, and structure them into a valid SQL query. The database schema is predefined, ensuring that queries align correctly with the stored data. Different query types,including retrieving specific records, filtering results, and performing basic aggregations are supported by the system design . The implementation ensures that the generated queries are both syntactically and semantically correct, reducing the chances of errors during execution.The system can efficiently adapt to various database structures while maintaining flexibility in interpreting user queries by following to predefined set of linguistic rules.

This literature survey explores existing research in speech recognition, natural language processing, and query generation techniques, analyzing their advantages and limitations. The effectiveness of rule-based query formation is examined in the study, highlighting its benefits in efficiency, interpretability, and reliability. Furthermore, it discusses the challenges associated with processing natural language input, ensuring accurate SQL query generation, and handling variability in spoken language. The rule-based approach is particularly beneficial in scenarios where domain-specific data constraints must be met, ensuring that only valid queries are generated. For applications with clearly defined query architectures, it is also a more

sensible option because it lower the requirement for big training datasets.

This literature survey explores existing research in speech recognition, natural language processing, and query generation techniques, analyzing their advantages and limitations. The effectiveness of rule-based query formation is examined in the study, highlighting its benefits in efficiency, interpretability, and reliability. Furthermore, it discusses the challenges associated with processing natural language input, ensuring accurate SQL query generation, and handling variability in spoken language. The rule-based approach is particularly beneficial in scenarios where domain-specific data constraints must be met, ensuring that only valid queries are generated. For applications with clearly defined query architectures, it is also a more sensible option because it lower the requirement for big training datasets. .

II. LITERATURE SURVEY

[1] The Voice-Based Natural Language Query Processing system is designed to allow users to interact with databases using spoken language without the need for SQL knowledge. The methodology consists of turning spoken queries into text, tokenizing and analyzing each term, and mapping to database terms in the dictionary. Syntactic and semantic analysis is then performed to create SQL queries, which are then executed on a relational database and results are displayed in a tabular format. The system also learns from user questions to become more accurate over time, though it still struggles with complex or unusual phrasing. Although the approach is user-friendly and appropriate for handling complex queries, it is constrained by ambiguities in natural language and reliance on predefined dictionary mappings, which may make it less accurate for diverse or complex phrasing.

[2] To meet big data related concerns, this study examines a number of data models, with a significance on models that manage high-volume, high-velocity, and high variety data.The study contrasts more modern NoSQL models, which provide improved scalability, flexibility, and horizontal distribution for massive data processing, with more conventional relational models. The major drawback includes complexity and heterogeneity of NoSQL models, which lack a common schema and frequently sacrifice consistency in favor of scalability and flexibility. Because of this it may be difficult to successfully integrate several data sources into a single framework.

[3] Human-computer interaction is enhanced by voice recognition and information retrieval systems , particularly for individuals with disabilities, by enabling speech-based access to digital libraries. For feature extraction technologies like Linear Predictive Coding (LPC) , Artificial Neural Networks (ANN) for keyword recognition using backpropagation, and Hidden Markov Models (HMM) for handling pronunciation variations improve accuracy. SQL manages the structured database for retrieval. However, variations in accents, speech rates, and background noise are the major challenges. ANN requires extensive training ,HMM struggles with continuous speech, and in noisy environments the system performs poorly , highlighting the need for further improvements.

[4] This system provide voice based natural language queries and simplifyed database interactions for common users. It help to convert speech into SQL commands using NLP techniques like tokenization, POS tagging, and keyword extraction. The process involves converting voice to text, parsing the query using a lexicon of SQL keywords and database attributes, and mapping keywords to database elements before generating an SQL syntax tree for execution. Stanford NLP tools enhance accuracy. However, the system is limited to basic SQL queries e.g., SELECT, WHERE, GROUP BY and lacks support for advanced features like joins and context-aware queries.

[5] The system uses Recurrent Neural Networks (RNNs) for audio-to-text conversion in a chatbot, enabling smoother interactions. It employs sequence-to-sequence models with Long Short-Term Memory (LSTM) to encode and decode audio, enhancing personalization by identifying user gender. Built with Keras and PyTorch, it integrates MySQL for data storage and Bootstrap 4 for the frontend, supporting deployment on platforms like Telegram. Limitations include a small training dataset affecting generalization, dependency on computational resources for scalability, limited optimization with Keras, and reduced adaptability beyond structured legal datasets.

[6] This research examines NL-to-SQL frameworks, with a particular focus on Natural Language Interface to Databases designed for users unfamiliar with SQL. Using the SPIDER dataset, it evaluates model accuracy and performance across various domains.

SQL-Net simplifies query prediction but struggles with handling complex queries. SyntaxSQL-Net improves accuracy by integrating syntax trees, while GrammarSQL ensures consistency through strict grammar rules. IR-Net use BERT to enhance related understanding but requires significant computational power. EditSQL refines queries based on context, and RAT-SQL, the most effective model, achieves more accuracy using relational-aware transformers.

Challenges remain in handling complex schemas and ambiguity. Future work may improve contextual understanding with ontological directed graphs.

[7] Information retrieval based on speech recognition results presents a selective retrieval method for large, untagged speech datasets like news broadcasts. Traditional methods struggle with recognition errors and high resource demands. This approach improves accuracy using a small distance matrix and a confusion matrix to handle common transcription errors. Using dynamic programming and the Julius speech recognition system, it aligns recognized text with queries, achieving six percentage extra accuracy than exact matching. However, it requires extensive preprocessing, high computational power, and further testing for different languages and dialects, limiting real-time applicability.

[8] This system seamlessly integrates IoT, fog, and cloud computing with a voice user interface powered by natural language processing . Designed to support underrepresented lan-

guages like Bulgarian, it enhances accessibility and functionality.The IoT-Fog-Cloud architecture combines IoT sensors, NLP models, and cloud services to process voice commands both locally and online. This hybrid approach boosts speed, security, and flexibility by reducing dependence on cloud processing. With speech-to-text and text-to-speech technologies, the system efficiently understands and responds to voice commands while keeping cloud usage minimal. Although effective for specific tasks, the system struggles with languages that lack NLP support, particularly tonal languages. Despite these challenges, it provides a practical voice-controlled smart home solution, bridging gaps in existing systems and making smart technology more inclusive.

[9] This study reviews 174 papers (2006–2018) on deep neural networks in speech-related areas like automatic speech recognition, emotional speech recognition, speaker identification, and speech enhancement. It analyzes factors such as datasets, languages, environments, feature extraction methods, evaluation metrics, and DNN models.

The findings shows a strong focus on automatic speech recognition, mostly using public datasets, MFCCs for feature extraction, and Word Error Rate for evaluation. Standalone DNNs dominate, with limited exploration of RNNs and hybrid models. Language diversity gaps, learning on neutral environments, and MFCC dependency are the main limitations . Future research should explore diverse datasets, hybrid models, and alternative evaluation methods for more broader applicability.

[10] Deep learning is plays an important role in text-to-SQL conversion, helps to connect between natural language and structured database queries. The encoder-decoder framework enables systems to interpret user queries and convert them into SQL, while schema linking ensures accuracy by connecting query terms with the correct database elements. Attention mechanisms further increases performance, especially when dealing with complex queries like joins and aggregations.

To improve accuracy and efficiency, various deep learning architectures are used, including CNNs, RNNs, Pointer Networks, and Transformers. These models depends on large datasets like Spider and WikiSQL for training, allowing them to better understand natural language inputs and generate precise SQL queries.

Despite these advancements, challenges remain. Handling nested queries, complex joins, and scalability continues to be a hurdle due to the high computational demands of deep learning models. Schema-aware models have improved accuracy, but further research is needed to make these systems more adaptable, expand dataset diversity, and enhance their ability to process complex SQL queries effectively.

III. METHODOLOGY

The Voice-to-Text Process in the system starts with capturing the user's voice input through a microphone. Then the system undergo preprocessing techniques such as noise reduction, normalization, and feature extraction to increase the accuracy. To convert spoken words into text efficiently the system utilizes the Web Speech API's 'webkitSpeechRecognition' ,ensuring lightweight and real-time processing.The next step is preprocessing the text to refine its structure by removing stopwords words using a manually defined stopword list, ensuring precise filtering tailored to SQL query formation. After preprocessing, the system moves to SQL Query Generation, where it extracts meaningful components using Natural Language Processing (NLP) technique like tokenization.It then identifies the user's intent such as retrieving or filtering data while mapping extracted terms to database entities through a predefined lexicon. This ensures accurate recognition of table names, column names, and values. The system includes a user-friendly, interactive interface where users can select a table from a dropdown menu. when user select a table, the attributes of that table are displayed to help the user in forming suitable questions. This feature helps users structure their voice or text queries based on the available attributes, ensuring accurate data retrieval.

The query generation process follows a structured approach. First, it extracts SQL commands like SELECT, WHERE, or ORDER BY. Then, it maps the extracted terms to the database schema before generating the final query using predefined SQL templates. Once the SQL query is generated, the system fetches the database, where it is connected to the SQLite database and executes the query. The results can be showed in two ways.one as audio output using the Web Speech API for text-to-speech and the other as text in a Django-based web interface. To simplify data entry, the system also supports CSV file uploads, allowing users to populate the database without manually inserting records. This improves database management and enhances usability. By integrating speech recognition and NLP-based SQL generation, the system provides an efficient, voice-driven database interaction platform. This makes it especially useful for users who are unfamiliar with SQL, offering a more accessible way to retrieve data.

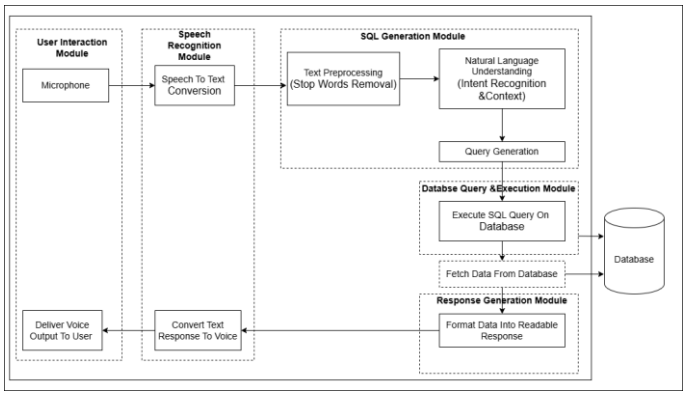

Fig. 1.System Architecture

The figure 1 illustrates the workflow of the voice command-based data retrieval system.Outlining its architectural structure and the sequential process of converting speech into SQL

queries, executing them on a database and retrive the data. The system is mainly divided into five modules, each one have a specific function.This ensure an efficient and structured approach to handling voice-based database interactions.

The User Interaction Module serves as both the entry and exit point of the system. It starts with the user providing a voice input through a microphone, which captures the spoken query. This voice input is then processed through different modules to extract relevant information and fetch the required data from the database. Once the information retrieval process is complete, the system ensures that the output is delivered back to the user in an accessible format.The system will display ouput in text format, also which converts the retrieved response into speech and delivers the voice output to the user, ensuring ease of use, particularly for those who rely on auditory interaction.

The Speech Recognition Module which convert spoken input into text. The first step in this module is Speech-to-Text Conversion, where the system processes the captured audio and convert it into a readable text format using webkit-SpeechRecognition API. In this step the unstructured voice input is converted into a structured textual form that can be processed further. Also when the SQL query has been executed and the database returns the requested information, this module also responsible for converting the textual response back into audio. The final output is delivered in an audio format, allowing users to receive the required information without need to read text on a screen.

Next is the SQL Generation Module which is responsible for transforming the processed text into an accurate and executable SQL statement. The first step in this module is Text Preprocessing, which involves removing stop words (such as ”the”, ”is”, ”of”) to clean up the input text. This step ensures that the system only processes meaningful words that contribute to the query’s intent. To determine the user’s intent (retrieving specific data) and extract relevant entities such as table names, column names, and filtering conditions from the text the system uses Natural Language Understanding techniques .Using rule based approach the system generates a structured SQL query that accurately represents the user’s request. This ensures that the generated query is both syntac-tically correct, making it executable on the target database.

The Database query and Execution module that will execute the generated SQL query and retrive relevent data from the database.The Response Generation Module is responsible for structuring the retrieved database output into a readable and meaningful format before presenting it to the user.

Finally text-based response is generated, it is converted into voice output and delivered back to the user through the User Interaction Module. This ensures that the system remains accessible to users who prefer voice-based interactions.

IV. Results And Discussion

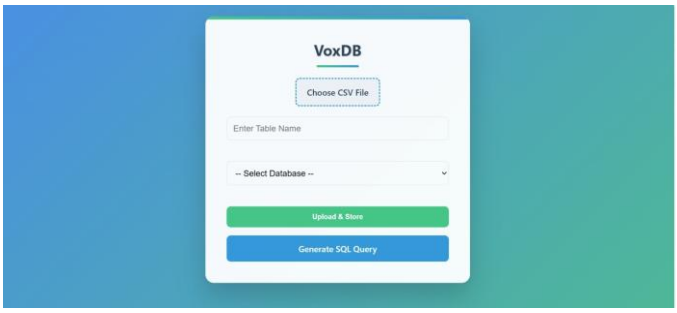

Fig.2 Home

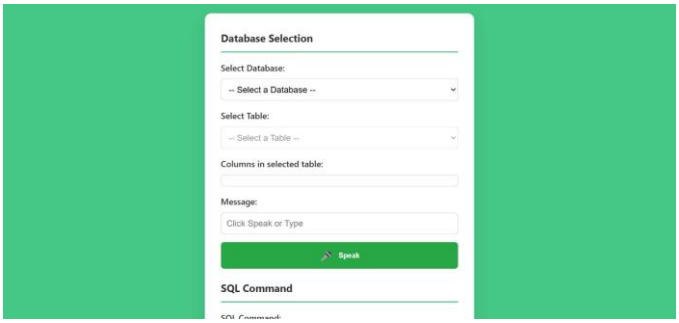

Fig.3 Database Selection Page

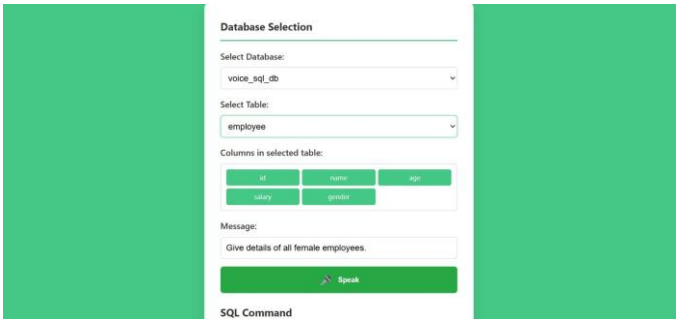

Fig.4 Voice Input

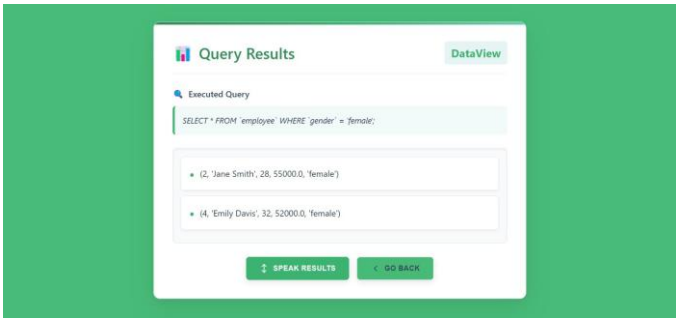

Fig.5 Query Generation

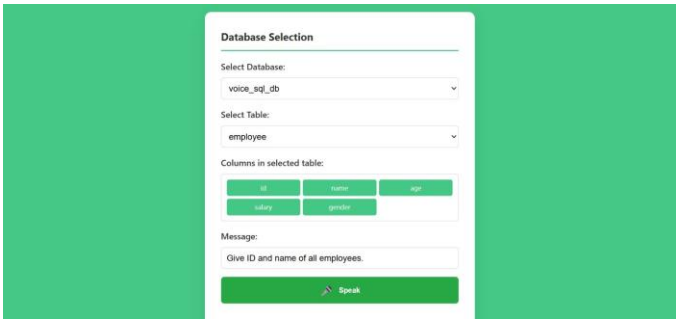

Fig.6 Text Input

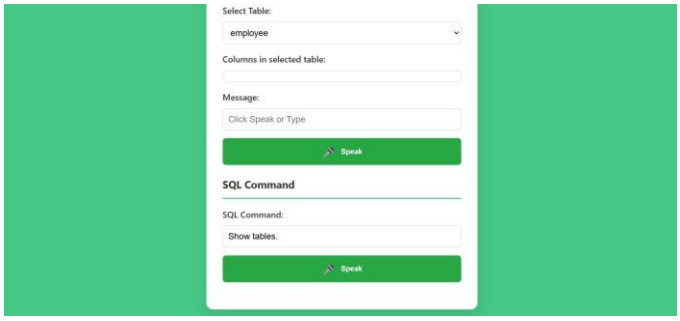

Fig.10 Show Tables

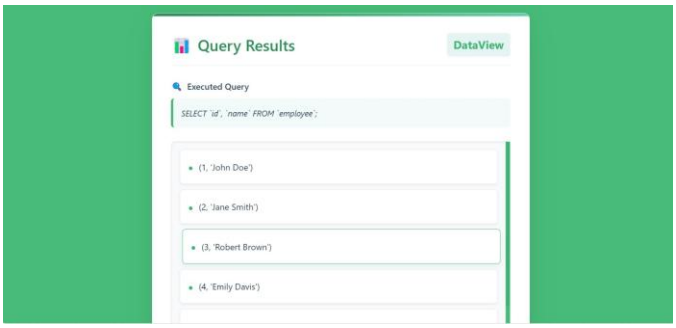

Fig.7 Query Generation

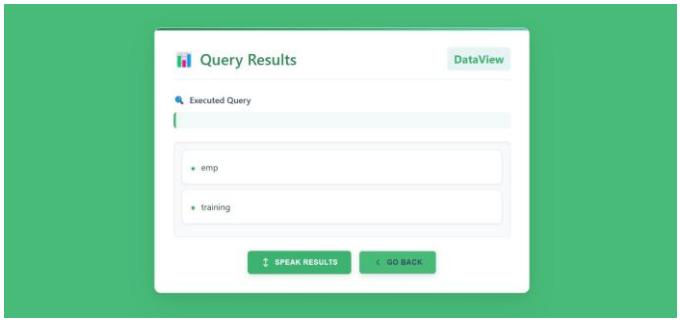

Fig.11 Retrived Results

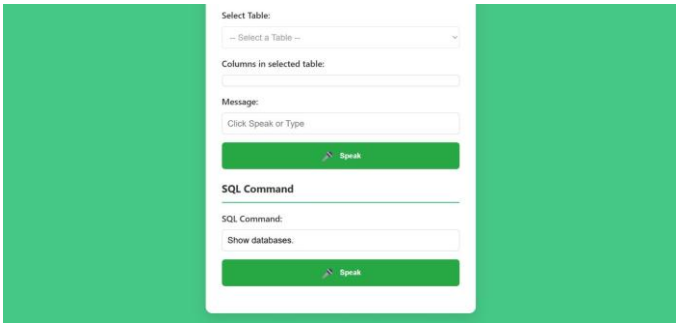

Fig.8 MYSQL Command

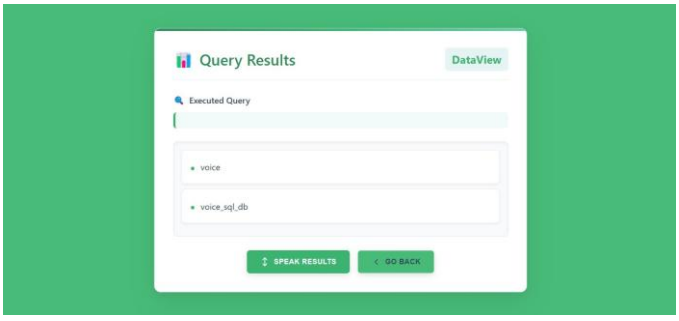

Fig.9 Result

Fig.2 shows the user interface,that allow users to upload a CSV file, specify a table name, select a database, and either store the uploaded data or generate an SQL query.This makes the data entry more easier.

Fig.3 dispalys a database selection interface that allow users to select a database,choose a table from the corresponding database and also it dispalys the columns of the selected table.It includes a text input field for user queries and a speak button, likely enabling voice-based input.Fig.4 depicts the implementation of this interface. And there is also an input field to execute the MYSQL commands.

Fig.5 shows the query generated for the user input,the query is executed on the database and the result is displayed.

Fig.6 illustares the generation of another different query,where a simple natual language input from the user side and fig.7 depict the corresponding query generated.

Fig.8 and Fig.10 depicts the implementation of MYSQL commands,when the user gives a mysql command as voice input or in text format.Fig.9 and Fig.11 shows the result of the corresponding commands.

V. CONCLUSION

In order to improve user accessibility ,voice command-based data retrieval systems have evolved significantly, integrating speech recognition, NLP, and SQL query generation. Our system streamlines this process by converting voice inputs into structured SQL queries and delivering results in an auditory format. Unlike cloud-dependent methods, it prioritizes local-ized processing, improving security and reducing latency.

The solution improves flexibility and customization in speech recognition by eliminating the need for the Google Cloud Speech-to-Text API. While it effectively processes queries and retrieves relevant database information, challenges remain in handling complex multi-table queries, expanding language support, and optimizing efficiency. Future improvements can focus on advanced deep learning techniques, better contextual understanding, and adaptive query generation to make the system more robust and scalable. Additionally, incorporating user feedback mechanisms can refine query accuracy and system adaptability. The integration of more diverse datasets and multimodal interaction support can further enhance usability. Ultimately, this system lays the groundwork for more inclusive and intelligent database interaction models.

VI. REFERENCE

[1] Munde, Puja, Sayali Tambe, Afreen Shaikh, Pratiksha Sawant, and Deepa Mahajan. ”Voice based natural language query processing.” Int. Res. J. Eng. Technol 7 (2020).

[2] F. Mostajabi, A. A. Safaei and A. Sahafi, ”A Systematic Review of Data Models for the Big Data Problem,” in IEEE Access, vol. 9, pp. 128889-128904, 2021.

[3] Shrawankar, Urmila, and Anjali Mahajan. ”Speech user interface for information retrieval.” arXiv preprint arXiv:1305.1429 (2013).

[4] Wahi, Varun, Avadhoot Joshi, Rohan Patel, Asad Mujawar, and Nishant Tayde. ”Voice Controlled Database Analysis” .

[5] Basystiuk, Oleh, Natalya Shakhovska, Violetta Bilynska, Oleksij Syvokon, Oleksii Shamuratov, and Volodymyr Kuchkovskiy. ”The Developing of the System for Automatic Audio to Text Conversion.” In ITAS, pp. 1-8. 2021.

[6] Baig, Muhammad Shahzaib, Azhar Imran, Aman Ullah Yasin, Abdul Haleem Butt, and Muhammad Imran Khan. ”Natural language to sql queries: A review.” International Journal of Innovations in Science Technology 4 (2022): 147-162.

[7] Watanabe, Masatoshi, and Masahide Sugiyama. ”Information retrieval based on speech recognition results.” Threshold 800, no. 1000 (2002): 1200.

[8] Iliev, Yuliy, and Galina Ilieva. ”A framework for smart home system with voice control using NLP methods.” Electronics 12, no. 1 (2022): 116.

[9] Nassif, Ali Bou, Ismail Shahin, Imtinan Attili, Mohammad Azzeh, and Khaled Shaalan. ”Speech recognition using deep neural networks: A systematic review.” IEEE access 7 (2019): 19143-19165.

[10] Kumar, Ayush, Parth Nagarkar, Prabhav Nalhe, and Sanjeev Vijayakumar. ”Deep learning driven natural languages text to SQL query conversion: a survey.” arXiv preprint arXiv:2208.04415 (2022).