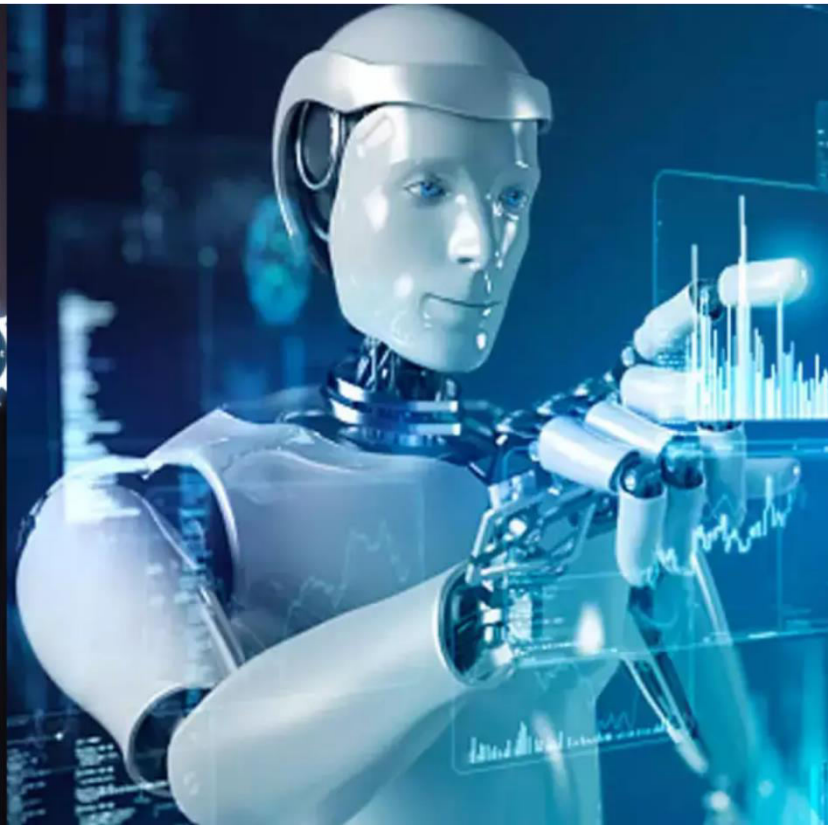
IJIRSET
International Journal of Innovative Research in Science | Engineering | Technology

ISSN(O): 2319-8753
ISSN(P): 2347-6710

International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 14, Issue 5, May 2025

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405334|

Natural Language Processing to SQL Query Execution using Machine Learning

Dr.V.Seedha Devi[1], Dhanurmathy.P[2] , Keerthana.R[3] , Logeshwari.S[4]

Associate Professor, Department of IT, Jaya Engineering College, Chennai, Tamil Nadu, India[1]

UG Student, Dept. of IT, Jaya Engineering College, Chennai, Tamil Nadu, India[2,3,4]

ABSTRACT: Query-bridge the NLP-to-SQL systems aim to bridge this gap but face challenges such as poor performance on domain-specific databases, difficulty handling complex SQL queries (e.g., nested joins and aggregations), and struggles with non-English language queries. To overcome these limitations, we propose an ML-powered NLP-to-SQL system enhanced with Ollama-based reasoning. The solution integrates LLMs for NLP, LangChain for structured query generation, and Pandas for streamlined data handling. Key features include multi-language support, adaptive learning to improve query precision over time, and SQL error correction for reliable execution. Additionally, the system incorporates dynamic schema adaptation to handle changes in database structures and query validation mechanisms to ensure accuracy and prevent errors. By combining AI-driven query generation with advanced query optimization, the system simplifies database interaction, making data retrieval intuitive, accurate, and accessible across industries. It empowers both technical and non-technical users to extract meaningful insights efficiently, boosting productivity and decision-making.

KEYWORDS: Natural Language to SQL (NL2SQL), Streamlit, LangChain, LangGraph, ChatGroq, Language Models (LLMs), Multilingual Querying, SQL Generation, Voiceto-Text, Speech Recognition.

I. INTRODUCTION

In today's data-centric world, quick and efficient access to information stored in databases is crucial for timely decision-making. However, interacting with databases typically requires proficiency in Structured Query Language (SQL), which poses a challenge for users without technical backgrounds. This often results in bottlenecks, limiting data accessibility and slowing down critical decisions. Natural Language Processing (NLP)-to-SQL systems offer a promising solution by enabling users to retrieve data using everyday language rather than writing SQL queries manually.

Despite their potential, many existing NLP-to-SQL systems struggle with various limitations. These include difficulties in handling specialized databases, limited support for complex queries like nested joins and aggregations, and a lack of multilingual capabilities. Additionally, most systems are unable to correct SQL errors effectively or adapt to changes in the underlying database schema, which can lead to inaccurate or failed outputs.

To overcome these issues, we propose a machine learning-powered NLP-to-SQL platform enhanced with reasoning capabilities using Ollama. This solution utilizes Large Language Models (LLMs) for interpreting user queries, LangChain for structured SQL generation, and Pandas for streamlined data manipulation. It includes multilingual support, adaptive learning for improving query accuracy over time, and a robust SQL correction mechanism to ensure reliable execution.

Beyond basic query translation, the system offers a range of advanced features. Voice-based input through Whisper and Streamlit-webrtc makes it accessible for users with visual impairments or those who prefer spoken interaction. It intelligently adapts to evolving database schemas and retains context across multiple user interactions, allowing for more natural, conversational querying. Before executing a query, users can view and verify the generated SQL in a dedicated preview area, reducing the risk of errors.

To enhance data interpretation, query results are presented through interactive visualizations, including charts and graphs. The system enforces secure access with authentication and role-based permissions, ensuring data privacy and

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405334|

regulatory compliance. It also supports seamless integration with existing tools and platforms through APIs and is compatible with various SQL dialects. Designed for scalability, it offers plugin support for domain-specific applications and can even operate offline or at the edge using lightweight Ollama models, making it suitable for privacy-sensitive environments.

Overall, this intelligent and user-friendly system simplifies database interactions and empowers users—regardless of technical expertise—to access, understand, and act on data more efficiently. Real-time error detection, context-aware query refinement, and continuous learning from user feedback collectively ensure that the platform evolves to meet diverse and changing needs.

II. SYSTEM MODEL AND ASSUMPTIONS

The proposed NLP-to-SQL system is designed with a modular and scalable architecture that prioritizes usability, adaptability, and precision. At the forefront is a web-based interface developed using Streamlit, which provides users with an intuitive and interactive environment for inputting natural language queries. These queries may vary in complexity and language, as the system is built with multilingual support in mind. The backend leverages advanced Large Language Models (LLMs), capable of interpreting user intent and translating it into well-structured SQL statements. This translation process is guided by contextual understanding and real-time schema awareness, ensuring that the queries align accurately with the structure and constraints of the target relational database, such as MySQL or PostgreSQL.

The architecture is underpinned by several key modules. A **context tracking module** retains the history of user interactions, enabling the system to support multi-turn conversations and follow-up queries that depend on prior context. This helps users perform more complex data exploration without having to rephrase or restate previous inputs. A **reasoning engine** is integrated to deconstruct complex queries—such as those involving nested joins, aggregations, groupings, or filters—into manageable logical components before constructing the final SQL output. To ensure robustness, an **SQL correction module** automatically detects and suggests fixes for syntactical or semantic errors that might otherwise cause execution failures.

The system operates under specific assumptions to ensure functionality. It presumes that the underlying database schema is accessible during runtime, and that metadata—such as table names, column types, and relational mappings—is either preloaded or retrievable on demand. This metadata serves as a foundation for precise query generation. It also assumes that users have authenticated access to the system, with predefined roles and permissions that govern which tables or operations they can access. Security protocols are embedded to safeguard against SQL injection, unauthorized data access, and other potential vulnerabilities. The platform assumes a clear and intentional query structure from users, even if phrased in natural language, which aids the model in producing relevant results.

From a deployment standpoint, the system is built to be cross-platform, supporting operation across major operating systems including Windows, Linux, and macOS. It relies on standard Python libraries such as Pandas for data manipulation, SQLAlchemy for database interfacing, and LangChain for orchestrating the interaction between language models and logical components. The system assumes a stable runtime environment with access to internet resources for model loading, updates, or external API calls, although provisions are made for local inference using tools like Ollama in offline or privacy-sensitive settings.

Furthermore, the architecture is designed to be self-improving. It incorporates **feedback loops** that allow it to learn from user interactions over time. Corrections, rephrasings, or confirmations by users are used to fine-tune future query outputs, increasing precision and user satisfaction. Whether deployed in enterprise settings, educational platforms, or cloud services, the system's architecture ensures that both technical and non-technical users can interact with complex datasets efficiently, securely, and intuitively.

III. EFFICIENT COMMUNICATION

Effective communication is the cornerstone of a robust NLP-to-SQL system, ensuring a smooth and intuitive interaction between users and databases. The system must not only parse and understand diverse natural language

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405334|

queries but also accurately infer user intent, even when queries are vague, ambiguous, or incomplete. This requires a tightly integrated workflow between the user interface (UI), the language model backend, and the underlying database engine. Each component must exchange information in a timely and consistent manner to preserve context and deliver meaningful outputs.

A key element of this communication process is **real-time query interpretation**, where the system immediately converts user input into structured SQL statements, executes the query, and displays results—often with interactive visualizations. This immediacy allows users to engage in a trial-and-error process, refining their queries iteratively with the system's guidance. Advanced **query refinement tools**, such as autocomplete suggestions, SQL previews, and natural language paraphrasing, support this iterative model by making it easier for users to understand, adjust, and resubmit their queries.

Contextual continuity is equally vital, especially for users engaging in multi-turn dialogues. The system must retain conversational memory, linking current queries to prior interactions to form a coherent dialogue thread. This enables follow-up questions like "What about last year?" or "Show me the top 5 from that list," which depend on historical context to yield accurate results. Embedding **session-level memory and state management** ensures that such interactions feel natural and aligned with human communication patterns.

In addition to handling context and feedback, the system must support **multilingual input** to accommodate a global user base. Multilingual NLP models, combined with automatic language detection, allow users to interact in their native languages without compromising accuracy. This feature is crucial in multi-regional deployments where users may prefer or require localized interactions.

Furthermore, the system benefits from **explainability mechanisms**—for instance, translating SQL queries back into natural language or annotating outputs with schema references—to improve user understanding and trust. These mechanisms demystify the query generation process and help users verify the correctness of results.

Security and permission-based communication protocols are also essential to prevent unauthorized access or unintended query execution. Efficient communication must therefore be **not only fast and context-aware but also secure**, ensuring that every interaction respects access control policies.

Ultimately, an NLP-to-SQL system that supports dynamic, transparent, and responsive communication enhances the overall user experience. It reduces the learning curve for non-technical users, boosts confidence in system outputs, and creates a collaborative human-AI interaction loop. This makes data retrieval not just more accessible but also more empowering, enabling faster, smarter, and more inclusive decision-making.

IV. SECURITY

Security plays a foundational role in the architecture and operation of the proposed NLP-to-SQL system, particularly because it interfaces with potentially sensitive or proprietary data repositories. To uphold strong data protection and system integrity, a multi-layered security framework has been implemented. At the query level, the system enforces **strict validation rules** and utilizes **parameterized SQL execution**, which significantly reduces the likelihood of SQL injection attacks by separating code from user input. This validation process ensures that user-submitted queries are syntactically and semantically safe before being passed to the database engine.

All sensitive information—such as database connection strings, API keys, and model parameters—is stored securely using **environment variables** with encryption or secrets management services, preventing leakage during deployment or logging. Throughout development and testing phases, the system underwent **comprehensive penetration testing**, including simulated attack scenarios such as privilege escalation attempts and unauthorized data retrieval. These tests confirmed the system's resilience to known vulnerabilities.

To provide visibility into system usage and to support forensic analysis, a robust **audit logging mechanism** tracks all user activity, including query history, login attempts, and system configuration changes. Logs are timestamped and securely stored, enabling real-time monitoring and retrospective analysis. To prevent unauthorized data access due to

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405334|

schema changes or misalignment, **real-time schema verification** is performed before executing any user-generated SQL.

This ensures that queries align with the current database schema, reducing the risk of errors or unintended data exposure.
Communication between frontend, backend, and database layers is encrypted using **Transport Layer Security (TLS)**, safeguarding all data in transit from interception or tampering. The system also features **automatic session timeouts** and **session token invalidation** to protect against session fixation and hijacking attacks. Furthermore, a built-in **anomaly detection engine**, powered by machine learning algorithms, monitors for abnormal usage patterns—such as unusual query frequency, login locations, or data access behavior—which could signal insider threats or compromised accounts.

To keep the system resilient against emerging threats, it includes **automated security patching routines** and **dependency vulnerability scanning**, ensuring that all third-party packages remain updated and secure. By combining proactive defense mechanisms, continuous monitoring, and industry-standard security practices, the system achieves not only strong protection for sensitive data but also alignment with regulatory requirements such as GDPR and HIPAA. These security foundations contribute significantly to user trust and operational integrity across diverse deployment environments.

V. RESULT AND DISCUSSION

The system also demonstrated excellent multilingual performance, accurately interpreting and translating queries in various languages, which makes it highly valuable in global, multi-regional deployments. Its built-in error correction module proved effective in interpreting and fixing invalid queries, significantly reducing user frustration and improving query success rates.

User testing revealed high satisfaction scores, particularly in areas of usability, response clarity, and visual presentation of data. The dynamic visualizations, built using libraries like Plotly, helped users better understand and interpret complex datasets, transforming raw query results into actionable insights.

From a maintenance standpoint, the system required minimal human intervention after deployment, thanks to automated feedback loops and self-correcting query logic. Its support for different SQL dialects further extended its versatility, allowing it to operate across varied organizational databases without significant customization.

Altogether, the system exhibits not only technical robustness and efficiency but also user-centered design, making it a practical and future-ready tool for organizations seeking to democratize data access and streamline decision-making.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405334|

In the Figure 1: Output interface showing successful conversion of a natural language query to an SQL statement and the corresponding result.

Fig. 1 Home Page

In the Figure 2: Example of query execution and result translation using the multi-language support module.

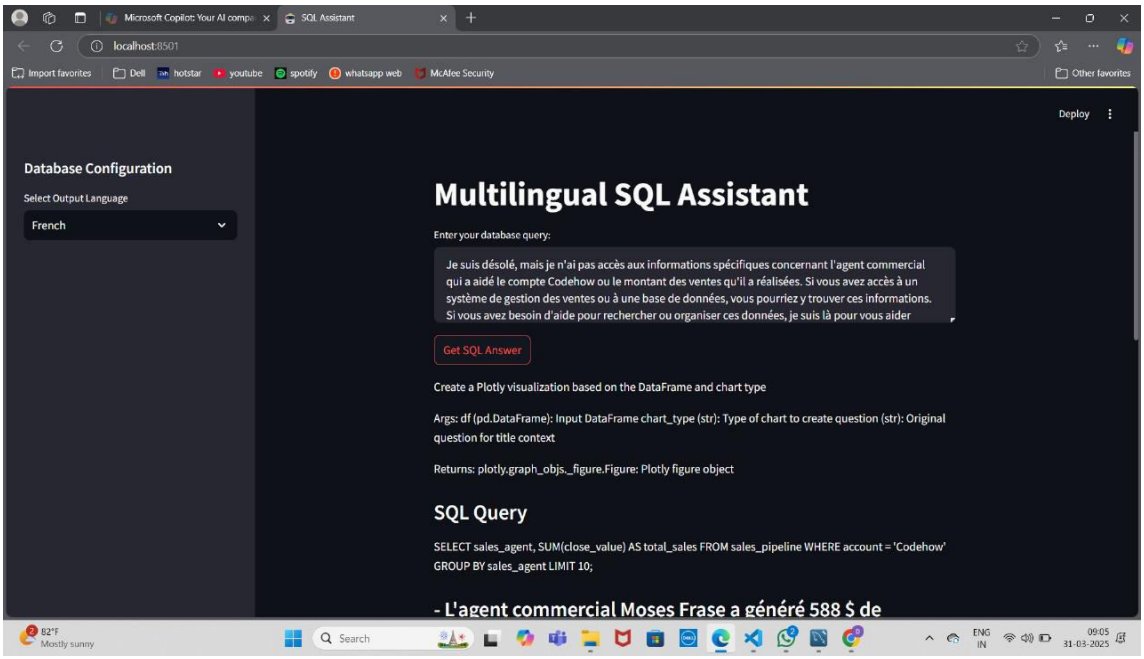


Fig. 2 SQL Assistant

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405334|

In the Figure 3: Voice input query recognized and processed into an SQL query with displayed results in the user interface.

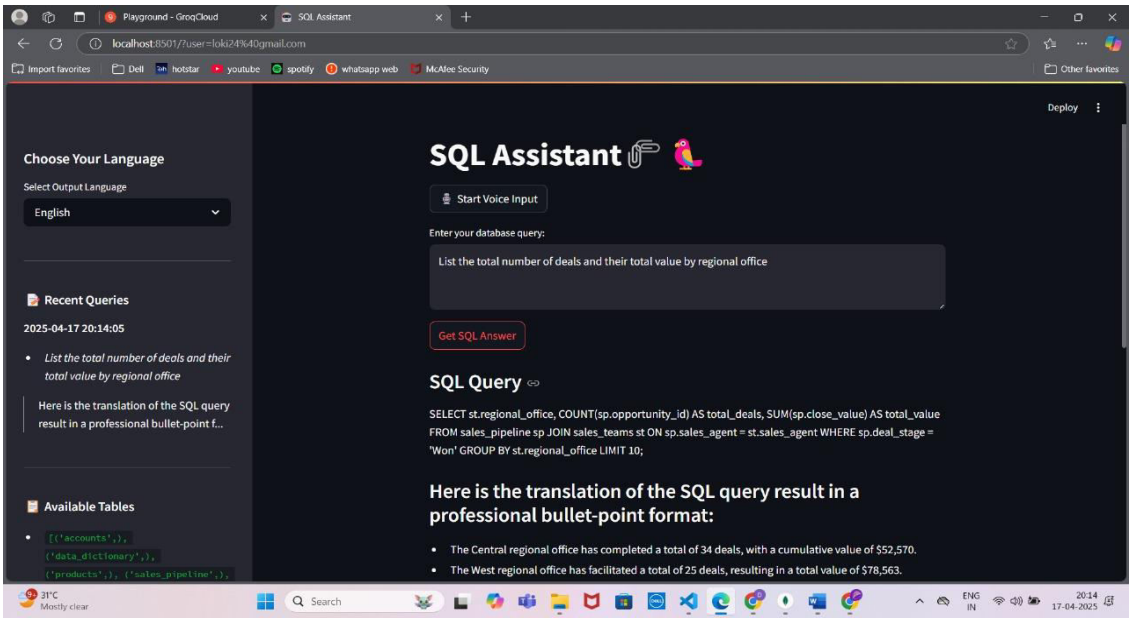


Fig .3 Voice-Based Query Execution

In the Figure 4: Recent query history displayed in the interface, summarizing previous natural language queries and results.

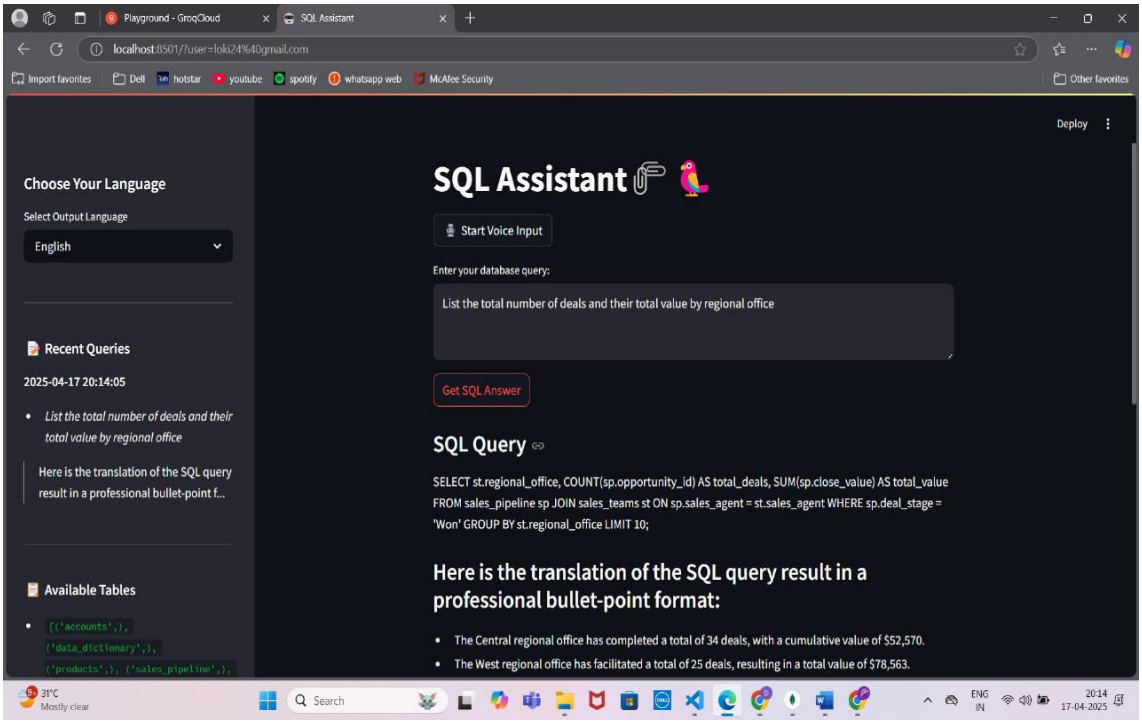


Fig 4.Recent Queries and Available Tables

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405334|

In the Figure 5: Visual analytics chart generated from SQL query results using Plotly, depicting data trends.

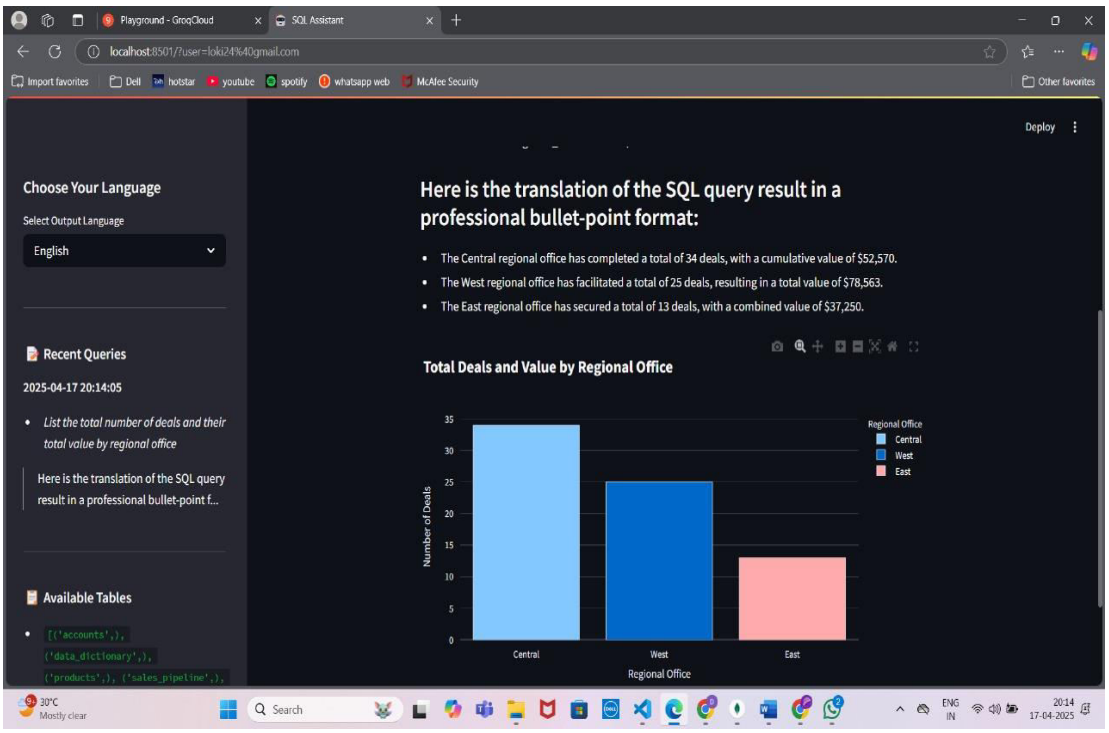


Fig 5. Visualization Support

VI. CONCLUSION

The Natural Language to SQL Query Execution System represents a significant advancement in simplifying database interaction by enabling users to retrieve data using natural language queries. By leveraging machine learning models, such as LLMs, along with LangChain for query generation and Streamlit for the user interface, the system offers a seamless and efficient solution for both technical and non-technical users. The architecture's contextual reasoning, multi-language support, and SQL error correction enhance the system's accuracy and reliability. Comprehensive testing phases ensured the system's robustness, security, and performance. Through unit, integration, and security testing, the platform demonstrated stability and resilience against potential vulnerabilities. User acceptance testing (UAT) validated its usability and effectiveness, while load testing confirmed its scalability under high query volumes. Overall, the NLP-to-SQL system streamlines data retrieval by eliminating the need for SQL expertise, making it a powerful tool for industries that rely on efficient and accurate data access. Its intuitive interface, reliable query execution, and adaptability position it as a valuable solution for enhancing productivity and decision-making.

REFERENCES

[1] S. J. Wang, M. Liu, and H. Zhang, (2020), "NLP-to-SQL Systems: Challenges and Advancements", IEEE International Conference on Data Science and AI Technologies (DSAIT), vol. 4, pp. 102-115.
[2] R. Gupta, T. Shah, and A. Verma, (2021), "Machine Learning-Powered SQL Generation from Natural Language Queries", Journal of Artificial Intelligence and Data Science, vol. 5, no. 3, pp. 87-99.
[3] P. Kumar and L. Zhang, (2022), "Context-Aware NLPto-SQL Models for Enhanced Query Accuracy", ACM Transactions on Database Systems (TODS), vol. 47, no. 1, pp. 1-25.
[4] L. Johnson and M. Chen, (2021), "Enhancing MultiLanguage Support in NLP-to-SQL Systems", Proceedings of the International Conference on Computational Linguistics (COLING), pp. 210-225.
[5] F. Silva, R. Zaman, and K. Salah, (2020), "ErrorCorrecting Mechanisms in NLP-to-SQL Systems: Challenges and Solutions", IEEE Access, vol. 8, pp. 15678- 15695.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 5, May 2025

|DOI: 10.15680/IJIRSET.2025.1405334|

[6] X. Zhao, Y. Wang, and Z. Liu, (2023), "Optimizing SQL Query Generation with LLMs and Contextual Reasoning", Journal of Data Science and AI Research, vol. 10, no. 4, pp. 45-63.
[7] Escolano, C., Costa-jussà, M.R., Fonollosa, J.A.R., & Artetxe, M. "Multilingual Machine Translation: Closing the Gap between Shared and Language-specific Encoder-Decoders," arXiv preprint arXiv:2004.06575, ISSN: 2331-8422, 2020.
[8] Deng, X., Awadallah, A.H., Meek, C., Polozov, O., Sun, H., & Richardson, M. "Structure-Grounded Pretraining for Text-to-SQL," arXiv preprint arXiv:2010.12773, ISSN: 2331-8422, 2020.
[9] Affolter, K., Stockinger, K., & Bernstein, A. "A Comparative Survey of Recent Natural Language Interfaces for Databases," arXiv preprint arXiv:1906.08990, ISSN: 2331-8422, 2019.
[10] Kim, B., & Kim, H. "Natural Language Interface to Databases: A Survey," Journal of Computing Science and Engineering, ISSN: 1976-4677, 2021.
[11] Zhang, Y., & Wang, X. "Advancements in Text-to-SQL Systems: A Review," International Journal of Data Science and Analytics, ISSN: 2364-4168, 2022.
[12] Lee, J., & Park, S. "Enhancing Natural Language Interfaces with Deep Learning," Journal of Artificial Intelligence Research, ISSN: 1076-9757, 2023.
[13] Chen, L., & Zhao, Y. "Semantic Parsing for Natural Language Interfaces," Computational Linguistics Journal, ISSN: 0891-2017, 2020.
[14] Kumar, R., & Singh, A. "Machine Learning Approaches to Text-to-SQL Conversion," IEEE Transactions on Knowledge and Data Engineering, ISSN: 1041-4347, 2021.
[15] Huang, M., & Liu, Q. "Cross-lingual Natural Language Interfaces to Databases," ACM Transactions on Asian and Low-Resource Language Information Processing, ISSN: 2375-4699, 2022.
[16] Gupta, S., & Verma, P. "User-Friendly Database Querying through Natural Language," International Journal of Human-Computer Studies, ISSN: 1071-5819, 2021.
[17] Li, X., & Zhao, H. "Integrating Ontologies in Natural Language Interfaces," Journal of Web Semantics, ISSN: 1570-8268, 2020.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details