IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Conversational Data Analysis

[1]Tejus Gupta, [2]Harsh Raj, [3]Abhishek Panwar, [4]Harsh Kushwah, [5]Ajay Kr. Shah

[1]Student, [2]Student, [3]Student, [4]Student, [5]Assistant Professor
Computer Science and Engineering (IoT)
Meerut Institute of Engineering and Technology, Meerut, U.P, India

Abstract: Accessing organizational data typically requires SQL knowledge, leaving non-technical staff dependent on IT. Our research presents an AI/NLP system that creates a natural language query interface. This allows users to access databases (PostgreSQL, MySQL, CSV) using simple commands. Using LlamaIndex and Gemini, it identifies intent, constructs SQL, and visualizes results. This framework improves accessibility and reduces IT dependency, filling the NL2SQL gap with an open-source, conversational solution for human-accessible information systems.

***Index Terms* -** Natural Language Processing (NLP), Artificial Intelligence (AI), Natural Language to SQL (NL2SQL), Data Query System, LlamaIndex, Google Gemini, FastAPI, Human-Accessible Information Systems)

Introduction

In the current day and age, information is what organizations rely upon for better decision-making, project planning, and performance analysis. Yet, when it comes to data retrieval systems, accessing such information requires SQL (Structured Query Language) and database management system endeavors.

Without technical expertise, the general population cannot query their information. This also limits the ease of access for organizational users. Consider the marketer, financial expert, business intelligence analyst, and C suite officer; if all these professionals need to engage developers and database administrators to get their answers, access will slow down operations, overburden teams, and prevent real-time insights.

Therefore, the goal of this project is to facilitate access with AI and NLP. Recently, NLP has hit the markets with AI prompting via powerful LLMs (Large Language Models) that comprehend natural language queries and can transform commands into structured approaches that databases, information systems, and BI truly understand.

The approach taken seeks to connect the gap between human language and computer language and the intentional patterns not created for natural use, but instead for a structured system.

While many systems today in BI boast NLP capabilities, they are either too big and expensive or too constrained with the lack of contextual engagement to fully satisfy end-user needs.

This proposal goes beyond other applications by creating an AI-based system that does everything from natural language processing to SQL generation to data retrieval and visualization. The insights this framework hopes to provide will lower barriers to accessing data and increase utilization for AI-driven commercial and research inquiry in research and industry.

I. LITERATURE REVIEW

2.1 *Evolution* of Data Query Systems

When systems began, structured query language (SQL) was the intended sense of a relational database language. Yet SQL requires users to possess a certain amount of technical skill to query successfully. Systems like MySQL Workbench and Oracle SQL Developer, Microsoft Access ease querying through GUI-enabled rendering, yet fail to open the same windows for those without SQL competency. Eventually, BI tools shortened access time to databases but also needed some training for those lacking technical skill sets to ensure accessibility - but at that same time, BI tools are intended to minimize access time to insight. This suggests an unnatural language interface for databases (NLIDB) that warrants further exploration.

2.2 NLP and AI Jump In

NLIDB use became an interdisciplinary field combining computational linguistics with NLP and AI to better facilitate user inquiry into data. Finegan-Dollak et al. (2018) noted that standardization is needed within the existing evaluative methodology for developer NL2SQL query intentions, while Yu et al. (2018) created Spider, the first large-scale complex database for text-to-SQL mapping operations. Machine learning, deep learning, and schema based representation emerged from findings mapping in-domain and out-of-domain intentions to help user understanding. Recent developments come from LLM - including OpenAI Codex, LlamaIndex and Google Gemini - which enable users to present natural language and generated SQL across domains for anticipated comprehension and results. Their findings use transformer-based architecture, contextual embeddings and pre-trained attention mechanism (ACM) to aggregate patterns for user intent without sacrificing relational integrity over time.

2.3 Visualization Makes Things Easier

Where these latest systems and access into BI systems take off is the automated features for visualization for greater impact on comprehension. Systems like Tableau, Powerbi, and more integrated BI sponsored systems facilitate visualization efforts without users knowing what they're necessarily doing.Metabase then takes this information and turns structured inquiries into graphs, charts, and dashboards. Research shows that visualization greatly increases understanding so that stakeholder data-driven business decisions can be made in a timely manner. Furthermore, it bridges the gap between such a retrieval and the actionable insights possible through AI-based requests and subsequent visualization since it essentially links the two together for natural acquisition.

2.4 But There Are Still Problems

However, current NL2SQL and BI solutions all have many disadvantages as well. For example, commercially available systems are either expensive and closed source or open source, but the open source options fail, bogged down by contextual failures, schema misunderstandings, and diminishing SQL success with bi- or tri-columned or nested asks. LLM options fall short, too, as they lack the real-time contextualization necessary for schema-dependent tasks or misstep if the correct schema isn't activated or context is lacking through multi-turn requests. All of these challenges mean there's a comprehensive need for a solid source, an open one with flexible, semantically nuanced power, and this incremental guide is just the start.

2.5 What This Research Is Doing

Natural language querying has been extensively studied in the field of query solicitation, but a comprehensive platform has yet to be created that combines NL2SQL as a reliable means of SQL creation in a single, open-sourced, user-friendly conversational engine that supports multi-database accessibility and automatic transition into visualization operation. This research contribution seeks to bridge that gap with an AI-Based Natural Language Data Query System that understands semantics through LlamaIndex and Google Gemini, and combines a FastAPI built for back-end development focused on scaling, efficiency, accuracy and user-friendliness for those without tech-savvy skills. This addition also subscribes to the growing trend of conversational analytics and other AI developments.

II. LITERATURE REVIEW

3.1 Research Problem:

With such developments in data analytics and artificial intelligence at every turn, the challenge remains for organizations that accessing data is anything but complicated and quick. The primary means of data querying requires complex structured query language (SQL) creation from those who possess a technical background, and developers or data analysts aren't always immediately available to satisfy simple asks. As such, companies fall behind in timeliness, dissemination of information, and effective use of data.

While current Business Intelligence (BI) solutions come with graphical modules to facilitate data rendering and visualization, they are often too costly, complicated, or narrowly created. While current Natural Language to SQL (NL2SQL) solutions - based on extensive language model applications - successfully render language into code, schema bridging, context understanding, and multiple database collaboration is still an issue. Thus, a stable, natural language transformation solution that reliably generates correct executable queries with easily understood results across multiple data sets, on a simple level, is nonexistent at scale.

3.2 Goals of the Study

The objective of the proposed study is to develop an AI-Based Natural Language Data Query System. To query databases in a natural, human-readable fashion without having to understand SQL, so one's queries can be put into action and visualized as results. The specific objectives are as follows:

1) *To create a framework that allows end-users to develop natural language queries that can be turned into SQL requests through Natural Language Processing (NLP) and machine learning applications.*
2) *To create a chat-type interface for easy data querying and access.*
3) *To render visualized results automatically into appropriate graph/table formats.*
4) *To support multiple database types for general (PostgreSQL and MySQL) and flat-file (CSV) databases.*
5) *To make a flexible, modular, and secure application that could eventually support future AI/integration into other components of a database.*

To make data information accessible to those without technical experience while maintaining query integrity and rendering clarity.

III. METHODOLOGY

This proposed system is AI based and will enable query processes with ease through natural language that will also visualize responses in an easier, conversational method. The proposed system architecture is detailed in the system architecture diagram that features three layers of communication: Frontend Web User Interface, Backend Service Layer, and Database Layer. Each layer functions independently and connectedly to provide a successful application from input to meaningful insight.

4.1 System Overview

The proposed system would render an AI-based natural language solution that does not require the user ever to know SQL commands for efficiently and effectively querying. Plain English inquiries would suffice and would be transformed into structured queries. These queries would then be executed on the connected databases and visualizations of the answers would be rendered.

4.2 Frontend: The User Part

Frontend Web Application built on React and Next.js serves as the user-facing layer. For example, when a user states, "Give me the total sales for last month" that's how they interface with the tool. It then sends secure HTTP requests to access and get these API services rendered in responses. For example, specific results will render as bar charts, line graphs, pie charts, etc. The look of the frontend must also be clean to appear as a conversational back and forth. Additionally, this layer needs to accommodate multiple turns in the dialogue so that one can get "the sales chart for last month" and then immediately get "show it only for delhi" with the same conversational path active**.**

4.3 Central Machine Layer: Backend Systems

Backend Services act as the operating engine for this solution. Thus, once the input is registered, the system operates as follows in order:

1) *Understanding Intent from User and Translating to SQL Query: Convert Natural Language → SQL Query operates through NLP models and machine learning techniques to determine intent from input, recognize entities within a sentence and construct a syntactically correct SQL command. As such, this model also considers schema mapping, relationships between tables, and aggregation logic for appropriate generation.*
2) *Executing Generated Query to Return Results: Execute SQL Query on Database component executes SQL statement upon generation. This layer connects to Database Layer either through PostgreSQL or MySQL or even flat-file databases in CSV format. The structured objects of returned results will then be passed on to the visualization pipeline.*
3) *Passing Results Back to Visualization Component: Results returned from Execute SQL Query module will be processed in Visualization Pipeline responsible for determining the type of result (using integer identification) and generating the best possible visualization (both from within the pipe and via Generate Appropriate Chart). The more flexibility of visualizations means easier understanding of results.*

4.4 Database Layer

This component provides effective structure to support obtaining and delivering structured datasets. As such, system can operate on various database types for additional versatility across platforms and upscaling potential. This layer may be expanded in the future to include NoSQL databases or even streaming datasets in real time for an industry-wide approach.

4.5 This System's Novel Contribution

What makes this system novel is its iterative feedback loop. Once users get their results, they can reiterate or ask a larger scope question, and the iterative loop retains the conversational context and effectively allows the refinements to be sent back to the NLP-to-SQL module. Thus, it's possible to have multi-turn questions. This implementation emulates real-life conversation and essentially makes it more usable and effective by reinforcing what's previously established.

4.6 The Benefits of Such Proposed Architecture

4.6.1 *Intuitive/Accessible: Empowers non-technical users to make intensive queries.*

4.6.2 *Modular/Extensible: New AI models or databases can be added.*

4.6.3 *Automated: No manual SQL creation or manual visualization creation is needed.*

4.6.4 Conversational: Visualization is immediately accessible and questions can be made in real-time based on the connected context

IV. COMPARATIVE ANALYSIS

This comparative research explores what's known in the field and performance gaps so that systems that understand Natural Language Interfaces to Databases (NLIDBs) and Business Intelligence systems and Conversational AI systems can be recognized to see where an AI-based Natural Language to SQL Query System can improve access, flexibility, and efficiency of use.

5.1 Comparison with existing NLIDB Systems

Older NLIDBs for databases like PRECISE, NaLIR, and ATHENA relied on rule-based parsing and template matching, often coming up short, though a great starting point for NLQ to SQL accomplishments when user input isn't overly convoluted or misinterpreted. These systems laid essential groundwork for successful NLQ to SQL but were limited in scope as they had to include necessary schema mapping first and could not easily pivot to other databases of varying schema.

Newer incarnations include text-to-SQL transformer models of T5 and BERT-SQL. Though these are impressive linguistically-based accomplishments of understanding, they're static in nature, without collaborative feedback and visualized output. Many rely on language-based prototype benchmark datasets like Spider and achieve fairly decent linguistic translation accuracy but are again, devoid of user-experience, visualized appeal.

The proposed system includes the ability to have a multi-turn conversation, schema adaptation, and automatic visualization. This is a missing bridge between NLIDB efforts in research and user-interactive application.

5.2 Comparison with Business Intelligence Tools

Existing Business Intelligence systems (BI Systems) such as Tableau, Power BI and Looker create visualizations based on backend data. However, they possess a wealth of data visualization and reporting properties that require the end user to understand the data model and how to create dashboards and how to visualize. In addition, they do not organically support natural language and boast an "Ask a Question" field that assumes datasets have been formatted ahead of time.

The system proposed is better because it supports natural language inherently and does not require predefined dashboards since one can create charts and summaries based on a conversation made in real time. This significantly reduces the learning curve and technical thresholds for layperson users.

5.3 Comparison with Conversational AI Systems

Conversational systems that exist today include ChatGPT, Dialogflow and Rasa, all of which inherently understand natural language input, but are not, themselves, inherently, connected with relational databases. They feature third-party integration efforts that enable access to the needed information but rely on integration that does not inherently transform user intentions into immediate SQL.

Feature	Existing NLIDB Models	Business Intelligence Tools	Proposed System
Query Input Method	Rule Based/Template	GUI-based	Natural Language
Adaptability to New Schema	Limited	Moderate	High
Visualization Support	Minimal	Extensive (manual)	Automated
User Interaction	Single-turn	GUI-driven	Multi-turn Conversational
Technical Skill Required	High	Moderate	Low
Scalability	Low–Medium	High	High
Cost & Accessibility	Academic prototypes	Commercial	Open & Scalable

Table 1: Summary of Comparative Findings

V. Discussion

The AI-Based Natural Language Data Query System provides an advanced level of accessibility for data analytics as it features Natural Language Processing (NLP) and Machine Learning (ML) in tandem with Database Management Systems (DBMS) to facilitate a new standard of effective querying for non-experts. Traditionally, querying a database requires some understanding of SQL which poses a threshold for decision makers and analysts without expert backgrounds. The proposed system eliminates this barrier with AI-based interpretation of human inquiries which translates to Database System constructs naturally.

From a system implementation perspective, the modularity of the architecture allows for distinct processes for language processing, generation, SQL creation, execution and delivery/visualization. This improves scalability and maintainability. In addition, the multi-turn conversational loop supports continuous corrections within the system which can clarify inquiries based on past context.

This corresponds with projected AI-based analytics within the industry where automation and personalization are combined for faster insights.

In addition, the LlamaIndex and Gemini integration offers different database arrangements and linguistic variations that are difficult to obtain with standard rule-based applications. This entails PostgreSQL, MySQL, and CSV databases, making it widely applicable in academic and professional settings.

This corresponds with projected AI-based analytics within the industry where automation and personalization are combined for faster insights.

In addition, the LlamaIndex and Gemini integration offers different database arrangements and linguistic variations that are difficult to obtain with standard rule-based applications. This entails PostgreSQL, MySQL, and CSV databases, making it widely applicable in academic and professional settings.

Yet the application has future limitations with query ambiguity, session context retention, and users' understanding of how the model works. These are important research areas, with companies taking special interest in explainability and transparency in ethical and accountable AI-driven solutions.

VI. Future Scope

1) The following considerations will improve this application for intelligent effectiveness, usability, and investigative capability:

2) Voice Recognition for Queries - Speech to text will one day allow questions (i.e. "List me the products with the best sales this year") to be asked verbally. This allows for a non-typing audience.

3) Contextual Multi-Turn Conversations - Multi-turn conversations will be maintained long-term in future capabilities. The application will understand follow-up questions such as, "Now show me that same result for last year".

4) Enhanced Visual Dashboards - AI-influenced visual analyses will comprise suggested charts or real-time dashboards while the user tries to assess the presented data.

5) Cloud Platform Compatibility - SaaS capabilities will promote real-time scaling among teams without implementation and integration of third-party APIs (i.e., CRM and ERP systems) for accessibility.

6) Greater Data Security and Governance - Role-based authentication, encryption, and audit trails will meet privacy and regulatory standards in industrial settings.

7) Hybrid Intelligence with Knowledge Graphs - Structured queries will be combined with knowledge graphs for semantically relevant reasoning about entities and relationships beyond standard database circumstances.

VII. Summary

The architecture of the implemented system shows how AI, NLP and Database Systems can all function in collaborative safety as the presented BlackBox allows for a seamless transformation of natural language for data-driven results. Therefore, developments with context awareness, multimodal input and voice-enabled, cloud-based analytics suggest such a system as the future of enterprise BI and educational analytics systems.

AI-Based Natural Language Data Query System is an innovation for modern data analytics. It brings a new connection between technically-inclined databases and non-technical users where Artificial Intelligence (AI) and Natural Language Processing (NLP) are concerned. Typically, SQL-based systems require users to convert their queries into a form that the database can process; however, this system allows users to directly engage with the database through dialogic engagement without having technical skills themselves.

This implementation also consists of multiple layers - from recognizing speech to generating SQL commands to executing the queries and automatically visualizing them - meaning that the most complex systems can be reduced to usable and even enjoyable experiences for all users to expand their potential. The same is true for the iterative AI models upon which this system relies - LlamaIndex and Gemini - allowing inter-database accessibility and query accessibility. Where traditional Business Intelligence (BI) systems and rule-based Natural Language Interfaces to Databases (NLIDBs) have shortcomings, this system finds a middle ground between transparency, extensibility and automation. An architecture that's as flexible as this one promise to be applied in all business, educational and research settings for real-world application and implementation.

Finally, the steps taken within this research pave the way for voice-enabled analytics, contextually-driven inquiries and SaaS cloud-based implementations. As today's society favors user-friendly AI programs and developments, the feasibility of this implementation champions this line of research as a worthwhile endeavor for extensive applicability among data access systems with intelligent foundations for use.

Acknowledgment

The author would like to express sincere appreciation to the project team and the IoT Department for their support and constructive contributions throughout the preparation of this review paper. Their assistance and technical insights have been valuable in improving the quality of this work. The author also extends special gratitude to **Mr. Ajay Kumar Sah** (Department of CSE-IoT) for his guidance, timely feedback, and continuous encouragement. His expertise and supervision played a significant role in shaping the direction and completeness of this paper.

References

[1] C. Finegan-Dollak, J. K. Kummerfeld, L. Zhang, K. Ramanathan, S. Sadasivam, R. Zhang, and D. Radev, “Improving Text-to-SQL Evaluation Methodology,” *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018.

[2] T. Yu, R. Zhang, K. Yang, M. Yasunaga, D. Radev, “Spider: A Large-Scale Human-Labeled Dataset for Complex and Cross-Domain Semantic Parsing and Text-to-SQL Task,” *Empirical Methods in Natural Language Processing (EMNLP)*, 2018.

[3] P. Rajkumar and R. Dinesh, “AI-Based Natural Language Querying for Databases,” *International Journal of Computer Applications (IJCA)*, vol. 185, no. 25, 2023.

[4] LlamaIndex Documentation, Version 0.10, 2024. [Online]. Available: https://docs.llamaindex.ai/

[5] Google Gemini Developer Documentation, Version 2.0, 2025. [Online]. Available:

[1] C. Finegan-Dollak, J. K. Kummerfeld, L. Zhang, K. Ramanathan, S. Sadasivam, R. Zhang, and D. Radev, “Improving Text-to-SQL Evaluation Methodology,” *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018.

[2] T. Yu, R. Zhang, K. Yang, M. Yasunaga, and D. Radev, “Spider: A Large-Scale Human-Labeled Dataset for Complex and Cross-Domain Semantic Parsing and Text-to-SQL Task,” *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2018.

[3] P. Rajkumar and R. Dinesh, “AI-Based Natural Language Querying for Databases,” *International Journal of Computer Applications (IJCA)*, vol. 185, no. 25, pp. 12–19, 2023.

[4] S. Zhong, Z. Li, and J. Chen, “A Survey on Text-to-SQL Models and Datasets,” *ACM Computing Surveys (CSUR)*, vol. 55, no. 10, pp. 1–36, 2023.

[5] Y. Gan, X. Zhang, and T. Yu, “Natural Language to SQL: Benchmarks, Methods, and Challenges,” *arXiv preprint arXiv:2109.03725*, 2021.

[6] A. Srinivasan, A. Khanna, and A. Kaur, “Conversational Interfaces for Databases: An Overview,” *IEEE Access*, vol. 9, pp. 159876–159891, 2021.

[7] LlamaIndex Documentation, Version 0.10, 2024. [Online]. Available: https://docs.llamaindex.ai/
[8] Google Gemini Developer Documentation, Version 2.0, 2025. [Online]. Available: https://ai.google.dev/gemini-api
[9] OpenAI, "GPT Models: Text Generation and Semantic Understanding," *OpenAI Developer Documentation*, 2024. [Online]. Available: https://platform.openai.com/docs
[10] M. Dong, J. Gao, and H. Liu, "Bridging Natural Language and SQL with Deep Learning Techniques," *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, vol. 36, no. 4, pp. 987–1001, 2024.
[11] A. Narayan and P. Mukherjee, "AI-Powered Data Analytics: A Review of NLP-Driven BI Systems," *IEEE Intelligent Systems*, vol. 39, no. 3, pp. 45–54, 2024.
[12] Tableau Software, "Ask Data: Conversational Analytics in Tableau," *Product Whitepaper*, 2023. [Online]. Available: https://www.tableau.com/ask-data