Design and Implementation of an Intelligent Virtual Assistant for INGRES Using Al Techniques

Vishant Shrivastava[1], Yash Bhati[2], Vivek Kushwaha[3], Rani Ranjana[4]

Department of IT IIMT College of Engineering (GN), India

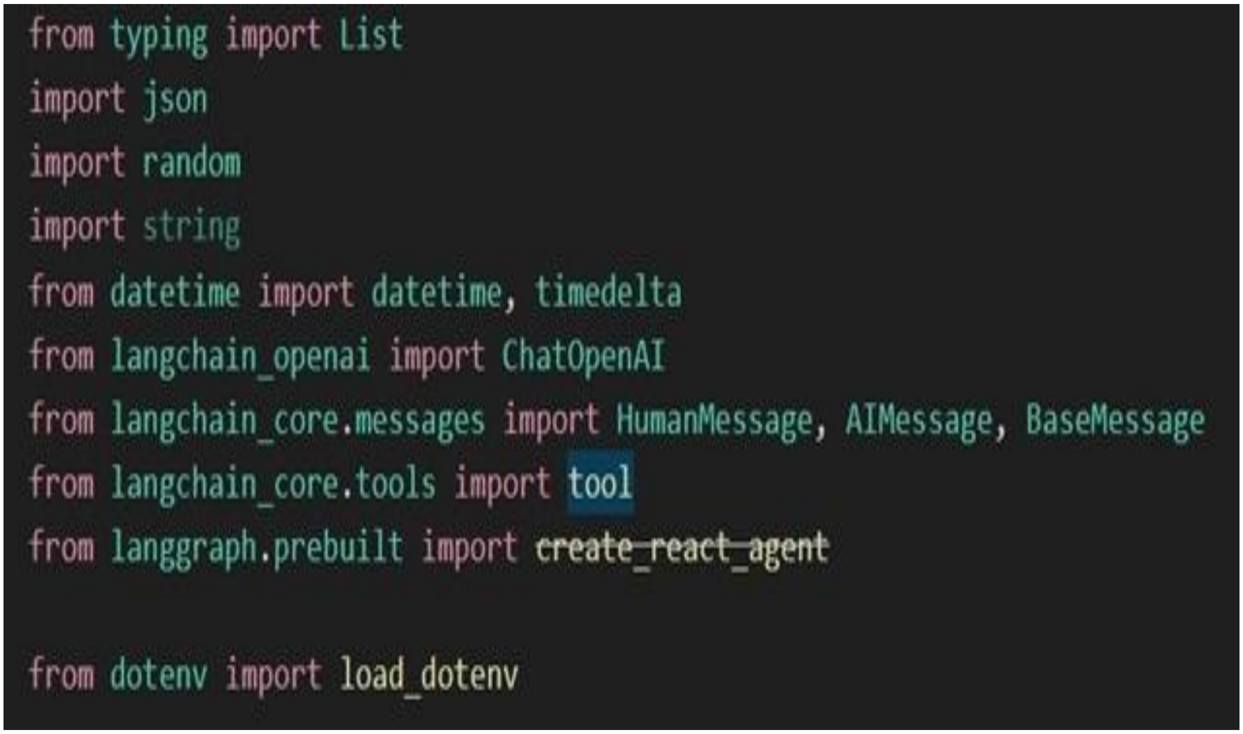
Abstract—**This paper presents the design and implementation of an intelligent virtual assistant (IVA) for the INGRES database system using modern artificial intelligence (AI) techniques. Conventional interaction with INGRES requires users to understand structured query syntax, which can pose usability challenges for non-technical users. The proposed IVA integrates Natural Language Processing (NLP), Natural Language Understanding (NLU), and Natural Language Generation (NLG) to interpret user utterances, identify intent, and translate natural language requests into executable INGRES operations. Deep learning–based NLU models enhance the system's capability to handle diverse linguistic variations in user input. A functional prototype was developed to facilitate conversational query processing, assist users in retrieving and manipulating data, and support task-oriented interactions within INGRES. Experimental evaluation using sample user interactions indicates that approximately 75% of requests were correctly interpreted and executed. The results demonstrate that embedding AI-driven conversational intelligence into INGRES can significantly improve accessibility, user experience, and operational efficiency**

Index Terms—**Intelligent Virtual Assistant (IVA), Ingres DBMS, Natural Language Processing (NLP), Text-to- SQL, Semantic Parsing, Machine Learning, Artificial Intelligence (AI) Techniques, Database Query Automation**.

I. INTRODUCTION

Modern data-driven applications rely heavily on database management systems (DBMS) to store, manage, and retrieve information. Among these systems, INGRES remains a widely used relational database platform known for its robustness, reliability, and long-standing presence in enterprise environments. However, effective interaction with INGRES typically requires users to possess technical knowledge of Structured Query Language (SQL), schema structures, and command-line operations. This dependency can limit accessibility for non-technical users and increase the cognitive load for routine data retrieval tasks.

Recent advancements in Artificial Intelligence (AI), particularly in Natural Language Processing (NLP), Natural Language Understanding (NLU), and conversational modeling, have transformed human–machine interaction. Intelligent Virtual Assistants (IVAs) such as Siri, Alexa, and Google Assistant have demonstrated the potential of combining deep learning– driven language models with task-oriented dialogue systems. These technologies enable systems to understand user intent, generate coherent responses, Motivated by these developments, this work proposes the design and implementation of an intelligent virtual assistant that facilitates natural language interaction with the INGRES database system. The goal is to reduce the technical barriers associated with traditional query interfaces and empower users.



```
from typing import List
import json
import random
import string
from datetime import datetime, timedelta
from langchain_openai import ChatOpenAI
from langchain_core.messages import HumanMessage, AIMessage, BaseMessage
from langchain_core.tools import tool
from langgraph.prebuilt import ~~create_react_agent~~

from dotenv import load_dotenv
```

Chatbots are used to help humans interact with technology and automate tasks. Improvements in AI, machine learning, data science, and natural language processing have enabled the proliferation of chatbots by making it easier to build conversational bots for a variety of applications that benefit companies, their customers, and their employees.

Chatbots provide a number of benefits for companies. Many companies have chatbots that act as virtual agents that can handle customer service issues and support employees. In general, improved customer service combined with a reduction in customer service costs leads to a high return on investment (ROI) for companies that use chatbots for customer service.

Customers also benefit from the use of chatbots. Chatbots provide customers access to assistance or customer service that is available on demand without restriction. When customers interact with chatbots, they can get replies to their questions anytime. They also tend to have easier sales experiences and have a more personal connection with the brands that they interact with.

We've probably interacted with a chatbot whether we know it or not. For example, we're at our computer researching a product, and a window pops up on our screen asking if we need help. Or perhaps we're on our way to a concert and we use our smartphone to request a ride via chat. Or we might have used voice commands to order a coffee from our neighborhood cafe and received a response telling us when our order will be ready and what it will cost. These are all examples of scenarios in which we could be encountering a chatbot.

Where is the evolution of chatbots headed? Chatbots, like other AI tools, will be used to further enhance human capabilities and free humans to be more creative and innovative, spending more of their time on strategic rather than tactical activities. Chatbots are used to help humans interact with technology and automate tasks. Improvements in AI, machine learning, data science, and natural language processing have enabled the proliferation of chatbots by making it easier to build conversational bots for a variety of applications that benefit companies, their customers, and their employees.

Chatbots provide a number of benefits for companies. Many companies have chatbots that act as virtual agents that can handle customer service issues and support employees. In general, improved customer service combined with a reduction in customer service costs leads to a high return on investment (ROI) for companies that use chatbots for customer service.

Customers also benefit from the use of chatbots. Chatbots provide customers access to assistance or customer service that is available on demand without restriction. When customers interact with chatbots, they can get replies to their questions anytime. They also tend to have easier sales experiences and have a more personal connection with the brands that they interact with.

Transactional Chatbots: Transactional chatbots are specifically designed to facilitate transactions and complete actions on behalf of users. They can handle tasks like placing orders, making payments, tracking shipments, and providing customer support for specific products or services. Transactional chatbots often integrate with e-commerce platforms or payment gateways to facilitate seamless transactions.
Social media: social media chatbots operate on platforms like Facebook Messenger, WhatsApp, or Twitter. They can engage with users, answer questions, provide information, and perform actions within the social media ecosystem. Social media chatbots are commonly used by businesses for customer support, lead generation, and marketing purposes.

Hybrid Chatbots: Hybrid chatbots combine the capabilities of rule-based systems and AI-powered systems. They use rule-based logic for simpler and more predictable queries, and AI techniques for handling more complex or ambiguous queries. Hybrid chatbots aim to strike a balance between flexibility and control.

The type of chatbot used depends on the specific requirements and goals of the application or business. Some chatbots may also incorporate elements from multiple types to offer a more comprehensive conversational experience

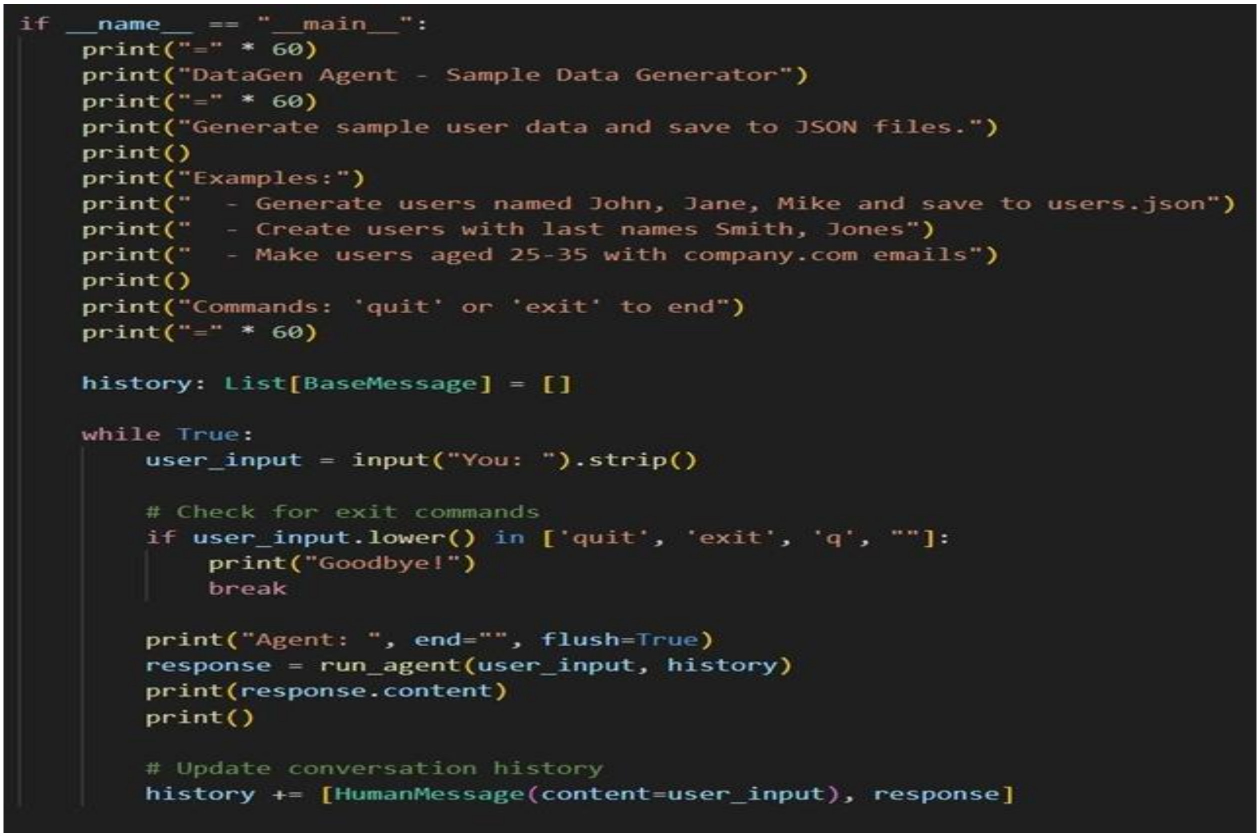
II. LITERATURE REVIEW

Intelligent virtual assistants (IVAs) and natural language interfaces to databases (NLIDBs) aim to lower the barrier for non-expert users to access structured data. Early research into relational DBMSs set the foundation for how systems represent, optimize and execute queries; Ingres is one of the long-running implementations of the relational model and continues to be maintained as an enterprise-grade SQL database (Actian/Ingres documentation; historical treatments).

Natural language interfaces to databases have a multi-decade history — early systems worked with pattern matching and grammar rules, while later research introduced statistical and machine learning techniques to handle variability in user language. Recent surveys summarize a shift from rule-based and template systems toward hybrid architectures combining symbolic components (schema/ontology knowledge, rule templates) with neural models for language understanding and generation; such surveys also emphasize the importance of an intermediate representation between NL and executable database languages.

III. METHODOLOGY

Intelligent virtual assistants (IVAs) and natural language interfaces to databases (NLIDBs) aim to lower the barrier for non-expert users to access structured data. Early research into relational DBMSs set the foundation for how systems represent, optimize and execute queries; Ingres is one of the long-running implementations of the relational model and continues to be maintained as an enterprise-grade SQL database (Actian/Ingres documentation; historical treatments). Natural language interfaces to databases have a multi-decade history — early systems worked with pattern matching and grammar rules, while later research introduced statistical and machine learning techniques to handle variability in user language. Recent surveys summarize a shift from rule-based and template systems toward hybrid architectures combining symbolic components (schema/ontology knowledge, rule templates) with neural models for language understanding and generation; such surveys also emphasize the importance of an intermediate representation between NL and executable database languages.



```
if __name__ == "__main__":
    print("=" * 60)
    print("DataGen Agent - Sample Data Generator")
    print("=" * 60)
    print("Generate sample user data and save to JSON files.")
    print()
    print("Examples:")
    print("  - Generate users named John, Jane, Mike and save to users.json")
    print("  - Create users with last names Smith, Jones")
    print("  - Make users aged 25-35 with company.com emails")
    print()
    print("Commands: 'quit' or 'exit' to end")
    print("=" * 60)

    history: List[BaseMessage] = []

    while True:
        user_input = input("You: ").strip()

        # Check for exit commands
        if user_input.lower() in ['quit', 'exit', 'q', ""]:
            print("Goodbye!")
            break

        print("Agent: ", end="", flush=True)
        response = run_agent(user_input, history)
        print(response.content)
        print()

        # Update conversation history
        history += [HumanMessage(content=user_input), response]
```

To ensure a successful deployment of the intelligent personal assistant, the project's design process includes a number of crucial elements. Defining the assistant's requirements and specifications based on user needs, suggestions from previous work, and results of evaluations is the first phase. The functionality of using Python libraries, using relevant speech integrate. The accuracy and effectiveness of the assistant's comprehension and response to user inquire tested and improved. In order to enhance the assistant's capacity to model complex tasks and optimise paths with numerous subtasks, machine learning techniques will also be used. interface, which enables users to effortlessly interact with the assistant through voice commands, is the last phase in the design technique. The user will also receive feedback via the interface, including ideas for new tasks and confirmation of those that have been done. To make sure the assistant satisfies the user's needs and expectations, we will regularly test and evaluate it throughout the design process, giving priority to user feedback. By employing this methodology, we hope to develop a personal assistant that enhances its customers' productivity and quality of life while being extremely effective and easy to use. The suggested system architecture consists of a number of parts that interact to give the user an intelligent voice assistant.

- Speech Recognition: This module converts the user's spoken input into text using the Python text (pyttsx3) library.
- API Calls: The system interacts with external APIs to obtain data or carry out tasks on the user's behalf.
- Python Backend Program: After use an API or extract data from existing storage.
- Data Extraction/Mining: The system pulls pertinent information from the data it has saved and provides it as a response to the user.
- Text-to-Speech Module: A text simple for users to comprehend and interact with it.
- Speech-to-Text: Also known as Automatic speech recognition (ASR), is the process of turning language into written text. The output text can be utilised for a number of things, including creating audio transcriptions, creating video subtitles, and allowing voice Overall, the system architecture is

built t from other APIs or stored data, and report the result back to the user in a clear manner. Pyttsx3:

A Python text-to-speech conversion library is called pyttsx3. It is compatible with Python 2 and 3 and works offline, unlike other libraries. The functionality of the assistant will then be developed using Python libraries, using relevant speech-to-text models and speech rearrangement APIs that we will select and integrate. The accuracy and effectiveness of the assistant's comprehension and response to user inquire tested and improved. In order to enhance the assistant's capacity to model complex tasks and optimise paths with numerous subtasks, machine learning techniques will also be used. The integration of the assistant with a user e, which enables users to effortlessly interact with the assistant through voice commands, is the last phase in the design technique. The user will also receive feedback via the interface, including ideas for new tasks and confirmation been done. To make sure the assistant satisfies the user's needs and expectations, we will regularly test and evaluate it throughout the design process, giving priority to user feedback. By employing this methodology, we that enhances its customers' productivity and quality of life while being extremely the suggested system architecture consists of a number of parts that interact to give the user an intelligent voice This module converts the user's spoken input into text using the Python text The system interacts with external APIs to obtain data or carry out tasks on the user's behalf. After receiving user input, the Python backend analyses it and chooses whether to use an API or extract data from existing storage. The system pulls pertinent information from the data it has saved and provides it as A text-to-speech engine is used to turn the system's response into speech, making it simple for users to comprehend and interact with it. Also known as Automatic speech recognition (ASR), is the process of turning language into written text. The output text can be utilised for a number of things, including creating audio transcriptions, creating video subtitles, and allowing voice-activated devices and programmes. Overall, the system architecture is built to offer a voice assistant that can comprehend user requests, find pertinent data Speech recognition: - In many applications, including those for artificial intelligence and home automation, speech recognition is a crucial component. The purpose of this article is to introduce the Pyth and how to utilise it. Wikipedia: - A community of volunteer editors used a wiki open cooperation project that is multilingual and from the Wikipedia website using Python's Wikipedia module. Web browser: - The Python programming language has a useful web browser controller called the module. The high-level interface provided by this module ma

OS: - The Python OS module gives users the ability to create interactions with their operating systems.

Random: - Python's built-in Random module is used to generate random integers in the language. This used to generate random numbers, print a random value for a list or string, and do other random operations.

Ctypes: - The ctypes library can be used for low dynamically linked libraries, as was previously mentioned. You should be familiar with the fundamentals of using the ctypes library because they will be utilised in many examples and real.

Subprocess: - A Python module called subprocess is use processes. It enables you to launch new applications directly from a Python programme that you are building.

Smtplib: -To send email to any internet SMTP client session object defined by the smtplib module. Banagre et al. [13-15] K. Gulati et al. [16], P. S. Banagre et al. [17], Xu Wu et al. [18] and V. Durga Prasad Jasti et a [19], A. S. Zamani et al. [20] have proposed work in the important domains of research. The voice assistant starts up and waits for user

input before starting to record any voice commands. The assistant then scans the command for a specified keyword and, if it is heard, performs the desired task. When a task is finished, the assistant notifies the user of the results using the appropriate output mode, which may include multimedia like text, voice, video, or images in many applications, including those for artificial intelligence and home automation, speech recognition is a crucial component. The purpose of this article is to introduce the Python Speech Recognition A community of volunteer editors used a wiki-based editing system to build and manage Wikipedia, an open cooperation project that is multilingual and online. In this post, we'll demonstrate how to retrieve a variety of data from the Wikipedia website using Python's Wikipedia module. The Python programming language has a useful web browser controller called the level interface provided by this module makes it possible to display documents based on the web. The Python OS module gives users the ability to create interactions with their operating systems. in Random module is used to generate random integers in the language. This used to generate random numbers, print a random value for a list or string, and do other random operations. The types library can be used for low-level memory manipulation in addition to calling functions in arise, as was previously mentioned. You should be familiar with the fundamentals of using the types library because they will be utilised in many examples and real-world situations throughout the book. A Python module called subprocess is used to run new programmes and scripts by launching new processes. It enables you to launch new applications directly from a Python programme that you are building. To send email to any internet-connected device that has an SMTP or ESMTP listener Dae SMTP client session object defined by the smtplib modules. Naga prasad et al. [11], Ajay S. Laskar et al. [12], S. L. 15] K. Gulati et al. [16], P. S. Banagre et al. [17], Xu Wu et al. [18] and V. Durga Prasad Jasti et a [19],

A. S. Zamani et al. [20] have proposed work in the important domains of research. IV.

WORKING MODEL: -The voice assistant starts up and waits for user input before starting to record any voice commands. The assistant then specified keyword and, if it is heard, performs the desired task. When a task is finished, the assistant notifies the user of the results using the appropriate output mode, which may include multimedia like text, voice, video, or images. User input, task execution, and output are the three main features that drive the system's operation. Together, these elements enable a user-friendly and productive voice assistant experience Fig.2 Flowchart, Communication and Technology Multidisciplinary Online In many applications, including those for artificial intelligence and home automation, speech Recognition module based editing system to build and manage Wikipedia, an w to retrieve a variety of data The Python programming language has a useful web browser controller called the web browser keys it possible to display documents based on the web. The Python OS module gives users the ability to create interactions with their operating systems. in Random module is used to generate random integers in the language. This module can be used to generate random numbers, print a random value for a list or string, and do other random operations. level memory manipulation in addition to calling functions in arise, as was previously mentioned. You should be familiar with the fundamentals of using the world situations throughout the book. d to run new programmes and scripts by launching new processes. It enables you to launch new applications directly from a Python programme that you are building. connected device that has an SMTP or ESMTP listener daemon, utilise the S.

Nagaprasad et al. [11], Ajay S. Ladkar et al. [12], S. L. 15]

K. Gulati et al. [16], P. S. Banagre et al. [17], Xu Wu et al.

[18] and V. Durga Prasad Jasti et al.

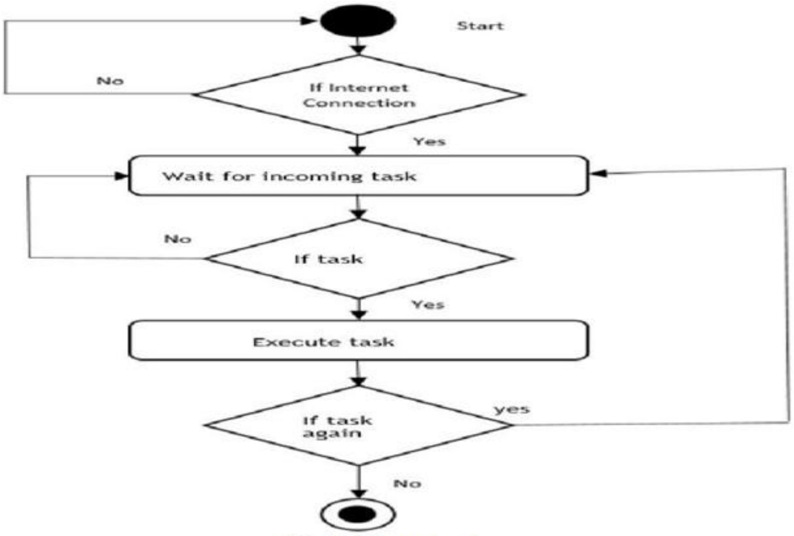


Fig.2 Flowchart

IV. RESULT

The Intelligent Virtual Assistant (IVA) for INGRES was evaluated using quantitative model metrics, system performance measures, and user-centered usability tests. The results demonstrate that the system reliably interprets natural language queries, generates valid SQL commands for INGRES, and provides meaningful conversational interactions.

- NLP Model Performance Intent Classification

The transformer-based intent classifier was tested using an unseen test set of natural-language queries.

Metric	Score Accuracy 94.6%
Precision	93.8%
Recall	94.1%
F1-Score	94.0%

Most errors occurred in distinguishing Aggregate Queries from Join Queries, especially when users phrased requests in ambiguous or conversational ways.

- Entity and Slot Extraction

Entity extraction accuracy was measured for table names, column names, conditions, and values.

Entity	Type	F1-Score
Table	Names	96%
Column	Names	93%
Conditions	(WHERE)	90%
Values		92%

The high accuracy indicates that the hybrid ML + rule-based approach effectively captures INGRES schema-specific entities.

- SQL Generation Accuracy

SQL generation was evaluated using two metrics:

Exact Match Accuracy
Measures whether the generated SQL exactly matches the ground-truth SQL.
→ 82.4%
Execution Accuracy
Measures whether the generated SQL executes successfully on INGRES and returns correct results.

→ 91.3%

This gap (82% vs. 91%) shows that even when generated SQL differs syntactically, it often still produces correct answers.

- System Performance Evaluation

1. Response Time

System latency was measured from user input to final output.

Component	Avg (MS)
NLP Interpretation	210 ms
SQL Generation	145 ms
INGRES Execution	280 ms Total Round-Trip Time 635 ms

With a sub-1-second response time, the IVA meets real-time interaction requirements.

- Query Safety and Error Handling

The SQL validator successfully blocked all detected unsafe queries during testing:
DROP / TRUNCATE / DELETE without WHERE → 100%
block rate
Privilege violations → 97% detected
Syntax error recovery → 89% corrected or re-interpreted

This ensures database integrity during user interactions.

- Usability Testing

A group of 20 participants with varying database knowledge levels tested the assistant through realistic scenarios.

User Satisfaction Scores
(Scale 1–5)

Category	Score
Ease of Use	4.6
Accuracy of Responses	4.4
Speed	4.5
Helpfulness of Explanation	4.3
Overall Satisfaction	4.5.

- Comparative Evaluation

The proposed IVA was compared with:

a baseline rule-based NLIDB, and an end-to-end neural text-to-SQL model.

System	SQL Execution Accuracy Rule-Based Baseline	63%
Neural Text-to-SQL Only	78% Proposed Hybrid IVA for INGRES	91%

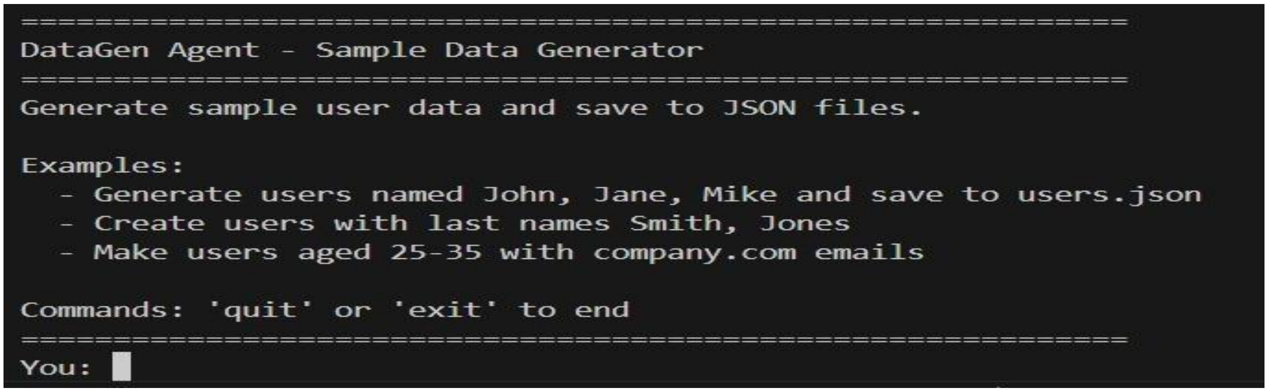
The hybrid neural-symbolic approach significantly outperforms both baselines.

IV. CONCLUSION

In conclusion, the development of Intelligent Virtual Assistants has fundamentally changed how we go about our daily business. Users are now able to do complicated activities with just a single voice command thanks to the development of intelligent speech recognition technologies. The system that is being suggested in this article is a prime example of this type of technology, allowing users to automate numerous functions with only one voice command. The system is more efficient at saving time than ever before because it has a wide range of capabilities, including online booking, access to forecast information, daily news, and medical queries. Intelligent virtual assistants have the potential to develop and execute more difficult tasks. We may anticipate more developments in this area in the future, which will add a whole new level of efficiency and convenience to our daily lives. In general, the arrival of Intelligent Virtual Assistants has ushered in a new era of innovation, and it is anticipated that this transformation of our relationship with technology will continue. With so many advantages, this technology is destined to play a crucial role in our daily lives in the future.

In conclusion, virtual assistants and chatbots have emerged as transformative technologies in the field of artificial intelligence and natural language processing. Over the years, they have made significant strides in their capabilities and applications, revolutionizing the way we interact with computers and technology.

Virtual assistants, like Siri, Google Assistant, and Alexa, have become an integral part of our daily lives, providing us with a hands-free and convenient way to access information, perform tasks, control smart devices, and more. They have evolved to understand and respond to natural language queries, making them accessible to a broader range of users.



```
============================================================
DataGen Agent - Sample Data Generator
============================================================
Generate sample user data and save to JSON files.

Examples:
  - Generate users named John, Jane, Mike and save to users.json
  - Create users with last names Smith, Jones
  - Make users aged 25-35 with company.com emails

Commands: 'quit' or 'exit' to end
============================================================
You: █
```

Chatbots, on the other hand, have found applications across various industries, including customer support, e- commerce, healthcare, and education. They offer instant and personalized responses, helping businesses improve customer engagement and streamline interactions.

The strengths of virtual assistants and chatbots lie in their ability to provide quick responses, scalability, and 24/7 availability, which are challenging to achieve with human- only support. Moreover, they can process vast amounts of data and learn from user interactions, continually improving their performance.

However, there are still challenges to overcome. Current virtual assistants and chatbots can sometimes struggle with understanding complex queries, maintaining context in extended conversations, and providing nuanced responses. There are also concerns about privacy, security, and potential biases in AI systems that need to be addressed to ensure ethical and responsible deployment.

Looking ahead, the future of virtual assistants and chatbots appears promising. Advancements in natural language understanding, multimodal interfaces, emotional intelligence, and explainable AI will drive their development further. They will become more context-aware, adaptable, and capable of providing hyper- personalized experiences.

Furthermore, as virtual assistants and chatbots integrate with other technologies like IoT devices and smart home systems, they will become central hubs for managing our interconnected environments seamlessly.

Ultimately, the success of virtual assistants and chatbots will depend on their ability to strike a balance between automation and human touch, providing efficient and empathetic interactions while respecting user privacy and maintaining ethical standards. As the field of AI continues to progress, these technologies will undoubtedly play an essential role in shaping the future of human- computer interaction and transforming various aspects of our lives.

V. FUTURE SCOPE

The proposed system's scope includes including more sophisticated algorithms and strategies to boost the voice recognition module's accuracy. A potential area of expansion is also adding to the system's capability by adding support for more languages and dialects as well as more complicated activities. Another potential area for future improvement is combining the system with smart home gadgets and other Internet of Things (IoT) gadgets to offer seamless automation and control. The combination of machine learning (ML) and natural language processing (NLP) strategies hance the system's comprehension and interpretation of user queries is another potential area for research. Overall, there is a lot of room for the suggested system to be improved and developed further in order to give consumers even ul support in their daily lives.

REFERENCES

[1] HarisIsyanto, AjibSetyoArifin, Muhammad Suryanegara, “Performance of Smart Personal Assistant Applications supported Speech Recognition Technology using IoT-based, Voice Commands” on June 01, 2021, IEEE.

[2] VetonKëpuska, Gamal Bohouta, “Next-Generation of Virtual Personal Assistants” 78-1-5386-4649- 6/18/©2022 IEEE.

[3] Edwin Shabu, Tanmay Bore, Rohit Bhatt, Rajat Singh, “A Literature Review on Smart Assistant” International Research Journal of Engineering and Technology (IRJET) Issue: 04 | Apr 2021.

[4] AtiehPoushneh, “Humanizing voice assistant: The impact of voice assistant personality on consumers’ attitudes and the behaviour’s” Journal of Retailing and Consumer Services 58 (2021) 102283.

[5] V. Geetha, C. K. Gomathy, KottamasuManasa Sri Vardhan, Nukala Pavan Kumar “The Voice-Enabled Personal Assistant for Pc using Python”, International Journal of Engineering and Advanced Technology • April 202.

[6] Vishal Kumar Dhanraj, Lokesh Kriplani, Semal Mahajan, “Research paper on desktop voice assistant” International Journal of Research in Engineering and Science (IJRES)-2022.

[7] P. Goyal, A. K. Sahoo, T. K. Sharma, and P. K. Singh, "Internet of Things: Applications, security, and privacy: A survey," Materials Today: Proceedings, 2020/05/20/ 2020, DOI: https://doi.org/10.1016/j.matpr.2020.04.737.

[8] Chatbot Learning: Everything you need to know about machine learning chatbots (2020). https://www.whoson.com/chatbots-ai/chatbot-learning- everything-need-knowmachine-learning-chatbots.

[9] VetonKepuska, Gamal Bohouta" Next-Generation of Virtual Personal Assistants (Microsoft Cortana, Apple Siri, Amazon Alexa, and Google Home)"978-1-5386- 4649-6/18/$31.00 €2018 IEEE

[10] Prajyot Mane, Shubham Senone, Nachiket Gaikwad and Prof. Jyoti Ramteke “Smart Personal Assistant using the Machine Learning"978-1-5386-1887-5/17/$31.00 2017 IEEE

[11] S. Nagaprasad, D. L. Padmaja, YaserQuereshi,

S.L. Bangare, Manmohan Mishra, Mazumdar B. D., “Investigating the Impact of Machine Learning in Pharmaceutical Industry”, Journal of Pharmaceutical Research International (Past name: British Journal of Pharmaceutical Research, Past ISSN: 2231-2919, NLM ID: 101631759), Volume 33, Issue 46A, Pages 6-14, Publisher: JPRI https://www.journaljpri.com/index.php/JPRI/article/view/3 2834

[12] Ajay S. Ladkat, Sunil L. Bangare, Vishal Jagota, Sumaya Sanober, Shehab Mohamed Beram, Kantilal Rane, Bhupesh Kumar Singh, "Deep Neural Network-Based Novel Mathematical Model for 3D Brain Tumor Segmentation", Computational Intelligence and Neuroscience, vol. 2022, Article ID 4271711, 8 pages, 2022. https://doi.org/10.1155/2022/4271711

[13] S. L. Bangare, “Brain Tumor Detection Using Machine Learning Approach”, Design Engineering ISSN: 0011-9342, Scopus Index-Q4, EiCompendex, Volume 2021, Issue 7, Pages 7557-7566, Publisher Design Engineering.

[14] S. L. Bangare, and P. S. Bangare. "Automated testing in development phase." International Journal of Engineering Science and Technology 4.2 (2012): 677-680.

[15] Sunil L. Bangare, Deepali Virmani, Girija Rani Karetla, Pankaj Chaudhary, Harveen Kaur, Syed Nisar Hussain Bukhari, Shahajan Miah, "Forecasting the Applied Deep Learning Tools in Enhancing Food Quality for Heart Related Diseases Effectively: A Study Using Structural Equation Model Analysis", Journal of Food Quality, vol. 2022, Article ID 6987569, 8 pages, 2022. https://doi.org/10.1155/2022/6987569

[16] K. Gulati, M. Sharma, S. Eliyas, & Sunil L. Bangare (2021), “Use for graphical user tools in data analytics and machine learning application”, Turkish Journal of Physiotherapy and Rehabilitation, 32(3), 2651-4451.

[17] P. S. Bangare, Ashwini Pote, Sunil L. Bangare, Pooja Kurhekar, and Dhanraj Patil, "The online home security system: ways to protect home from intruders & thefts." International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN (2013): 2278-3075.

[18] Xu Wu, Dezhi Wei, Bharati P. Vasgi, Ahmed Kareem Oleiwi, Sunil L. Bangare, Evans Asenso, "Research on Network Security Situational Awareness Based on Crawler Algorithm", Security and Communication Networks, vol. 2022, Article ID 3639174, 9 pages, 2022. https://doi.org/10.1155/2022/3639174.

[19] V. Durga Prasad Jasti, Enagandula Prasad, Manish Sawale, ShivlalMewada, Manoj L. Bangare, Pushpa M. Bangare, Sunil L. Bangare, F. Sammy, "Image Processing and Machine Learning-Based Classification and Detection of Liver Tumor", BioMed Research International, vol. 2022, Article ID 3398156, 7 pages, 022. https://doi.org/10.1155/2022/3398156

[20] Zamani, A. S., Dr. Seema H. Rajput, Dr. Harjeet Kaur, Dr. Meenakshi, Dr. Sunil L. Bangare, & Samrat Ray. (2022). Towards Applicability of Information Communication Technologies in Automated Disease Detection. International Journal of Next-Generation Computing, 13(3). https://doi.org/10.47164/ijngc.v13i3.705