American International Theism University Scientific Research Journal (AITUSRJ)
ISSN (E): 2578-3882**, ISSN (P):** 2578-3874

10.63094/AITUSRJ.25.4.1.3

Sigma - Virtual Assistant

Saba Imtiaz
Bahria University,
Karachi, Pakistan
sabaimtiaz.bukc@bahria.edu.pk

Ayesha Anees Zaveri
MIIT, Universiti Kuala Lumpur
Kuala Lumpur, Malaysia
zaveri.ayesha@s.unikl.edu.my

Furqan Ahmed
Bahria University,
Karachi, Pakistan
furqanxahmed11@gmail.com

Sehrish Habib
MIIT, Universiti Kuala Lumpur
Kuala Lumpur, Malaysia
sehrish.habib@s.unikl.edu.my

Izma Shehzadi
MIIT, Universiti Kuala Lumpur
Kuala Lumpur, Malaysia
Izma.shahzadi@s.unikl.edu.my

Muhammad Waqar Bari Awan
Department of Computer Science
University of agriculture Faisalabad, Pakistan
waqar.bari@uaf.edu.pk

***Abstract*—** This document presents Sigma, a virtual assistant with enhanced abilities for performing more complex actions. Sigma can search for any video on YouTube and play it for the user, launch other applications, chat, send WhatsApp messages, and respond to voice commands. Sigma handles the system commands and the interactions with the users through the PyAudio, pvporcupine and HugChat libraries, allowing him to manage the given tasks, recognize certain trigger words, and interact with users naturally. The project focuses on providing a friendly GUI and combining several modules to build a complete, easy-to-use virtual assistant.

***Index Terms* –** Virtual Assistant, Sigma, Hot word detection, Command execution, Media playback, Conversational responses, Natural Language Processing (NLP), Machine Learning (ML), Python, Voice recognition

I. INTRODUCTION

Integrating a new AI-powered virtual assistant could significantly boost the performance of a given system, especially against the backdrop of the new technological advancements today [1]. "It does not get any better than this! Sigma will allow you to configure and accomplish numerous tasks with voice commands effortlessly." [2]. Users can issue verbal commands to control media and application playback, web navigation, and a broad spectrum of other activities [3]. With command, execution can feel and exercise authority over their digital environments powered by formidable algorithms engineered to act upon user instructions [4]. Conversational command processing is another advancement added to this project, allowing [5]. This integration enables users to converse meaningfully and receive sound, verbiage, and information like humans. The user experience is smooth and hassle-free because of Sigma's tool and Python library integration [6]. Sigma acts with eloquence and sophistication as a revolutionary aid to day-to-day activities. Furthermore, adding competing performance variants aims at improving user satisfaction [7]. Data competent Business Intelligence Systems boost decision-making and operational effectiveness within the organization [8].

II. LITERATURE REVIEW

The improvement of AI and machine learning technologies have made virtual assistant worlds more accessible and preferred. Siri, Alexa, and Google Assistant are the frontrunners in personal digital assistance. Today's systems are no longer limited to following commands; they are sophisticated, context-sensitive, conversational systems. Assistants today are equipped with programmable natural language processing (NLP), machine learning (ML), and voice recognition, making the user experience nearly seamless. They comprehend the intricacies of the human language, adapt to the users' needs and perform highly accurately, efficiently, and quickly. From extensive studies in virtual assistance, significant data has been gathered regarding its foundational technologies. Deep learning models like RNNs and LSTMs, which depend on self-attention mechanisms, have revolutionized speech recognition and made understanding verbal information more reliable and precise. Together with the remaining voice recognition technologies, they provide a strong basis for system work done by users and virtual assistants. Comprehension, interpretation, and generation of any human language are made possible through natural language processing (NLP). NLP makes it possible for assistants to understand human language and respond appropriately. User inputs must be analyzed through tokenization, stemming, and named entity recognition. Chatbots and assistive NLP tools have machine learning algorithms to enable them to improve their efficiency and performance based on experience iteratively.

Enhancing dialogue user interface has culminated into a novel focus of research that studies the embedment of chatbot features. Through NLP and machine learning, contextually relevant retorts are generated by chatbots, and they respond to users in a more human-like interactive manner. Also, the advancement of systems such as SQL and NoSQL helps store and retrieve large amounts of information seamlessly while ensuring prompt and accurate answers are provided to user queries, which is augmented further with database management systems. Hot word detection, phrase detection and recognition

systems such as SVM and KNN have gained a prevalent focus in research and industry due to their significant role in user interaction. Their primary role is to enable user interaction as the Assistant remains in standby mode. The Assistant captures the user's voice input and performs the action set for that hot word or phrase.

Introducing new technologies, especially deep learning-based ones, leads to considerable advances in artificial intelligence (AI) and improves the TTS systems. Deep learning paradigms WaveNet and Tacotron are at the heart of the recent successes of imitating a human voice. They are capable of generating expressive speech [14]. Technical developments in TTS technologies to produce eloquent speech aid the user in receiving spoken responses that are readily understandable. The AI literature suggests applying new programming languages and frameworks while designing AI assistants. A set of Python libraries for speech synthesis, speech recognition, and even audio stream control offers modern solutions by enabling sophisticated frameworks for very simple dealing with streams in real-time and user monitoring. Other web technologies such as Eel, HTML5, CSS, and JavaScript provide simple yet effective means for the efficient and novel design of active and user-friendly interfaces guiding the user.

In the setting up of AI assistants, maintaining high-level security and privacy is critical [16]. Comprehensive studies underscore the need for adequate privacy protection and safeguarding of user data using encryption methodologies and structured access privileges. Building trust about sensitive data is one area of attention that requires utmost care. Feedback loops, adaptive learning, and personalization need to be incorporated to improve user engagement. These attributes increase responsiveness and intuitiveness, thus enabling better satisfaction of user preferences and needs. As noted in prior studies, there is also the need for multi-lingual and situational flexibility [18]. This allows the Assistant to function effectively in various situations and languages. AI system scalability is one issue which needs addressing. This includes the handling of huge datasets and ever-growing user capabilities. Additional assistance resources provided by the cloud, such as AWS, Google Cloud, and Azure, enable the provision of flexible heightening of services to aid proper assistant functioning and unattended smooth operation in other situations.

The upcoming features of AI assistants [20] indicate that they will soon include features such as identifying emotions, improved personalization, and interfacing with IoT devices. These trends point to more advanced, versatile, and connected virtual assistants capable of managing multiple activities and providing a richer experience to users. From executing system commands and launching applications to media playback and web scraping, Sigma handles it all. It exploits NLP, ML, and voice recognition technologies to accomplish these tasks. By implementing a chatbot, Sigma enhances conversational interaction, transforming user experience into a delightful and captivating one. The extensive review of prior work deeply informs the design and construction of Sigma. It provides a wealth of knowledge on the subject.

The aim is to create an assistant that, in addition to satisfying the user's needs, goes further in expectations, demonstrating the potential of sophisticated AI in personal digital assistance.

III. PROBLEM STATEMENT

Developing a virtual assistant that understands and executes various user commands involves multiple technologies for seamless function and user experience. "Sigma", our virtual Assistant, is specially configured by us to perform multiple tasks to further improve user experience with the added degree of sophistication and intelligence of the Assistant.

As shown in Figure 1, Sigma possesses the following features:

- Recognition of Hot Words for Activation: Sigma can recognize hot words that activate the Assistant. This ensures that the Assistant is always ready and can promptly serve user requests, so hands-free mode is highly convenient and effective[16].
- Executing System commands: Sigma can execute multiple system commands. From opening applications to changing system preferences and performing other routine activities, Sigma effortlessly handles these requests. This feature aims to streamline the user's work process and enhance productivity [3].
- Enjoy YouTube Video and Music Playback: Sigma permits users to stream their favourite videos and songs available on YouTube with a few voice commands. This feature serves as a form of entertainment. It is beneficial for accessing educational videos, tutorials, or other content [19].
- Improving Conversational Response with Chatbot Integration: Sigma has an incredibly advanced chatbot that automatically provides conversational replies. This chatbot is designed to understand and respond to various questions, providing pertinent, insightful, and interesting responses [5]. Utilizing natural language processing (NLP) technology ensures efficient and effortless interaction with Sigma.

With the help of modern Python libraries and other sophisticated resources, Sigma integrates these features to form a coherent and effective virtual assistant [6]. The intent is to build a practical and adaptable assistant, providing intelligent interaction with the user's digital environment. Through the integration of hot word detection, execution of system commands, media playing, and conversational interactions, Sigma aims to change the

user experience for the better and set a new standard for virtual assistants [7].

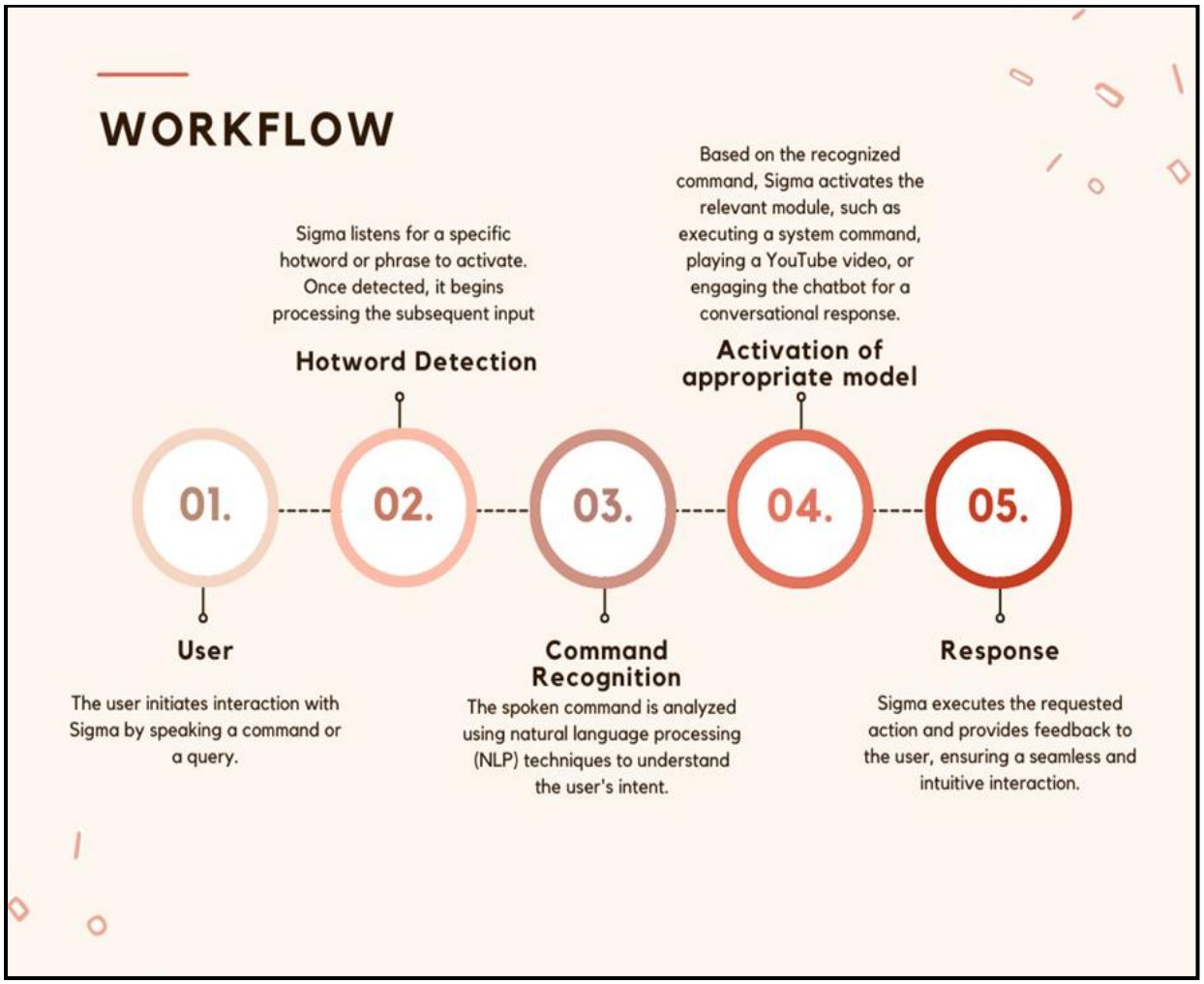

Figure 1 Overview of the Project

IV. Objective and Goals

- Construct the system to perform numerous exercises like creating a Virtual Assistant [3].
- Assistant to be capable of playing YouTube videos [19].
- Implement the capability to listen for and respond to specific keywords [16].
- Add a conversation chatbot to answer questions posed by the users [5].
- User Interface: Ensure that navigating to and engaging with the Assistant is intuitive and seamless [7].
- Considerable and steady completion of defined functions by the Assistant is a priority. Thus, Dependability and Functionality must be ensured [8].

V. Project Scope

The aim of the given project is to:

- Design a smart assistant that interacts with the operating system and the internet. The research paper highlights the precise command execution aspects [3].
- Looking into dialogue-based interactions [5].
- Facilitating YouTube videos to be played [19].
- Recognition and response to activation triggers [16].

The main focus of the research paper is to develop a user interface for a voice-activated computer for users with mobility and vision impairments while maintaining the functionality and reliability of the Assistant [7][8].

VI. Overview of the Project

The project overview can be seen in Figure 1 and Figure 2.

A. *Software*

- The software utilized in the project is:
- The Assistant is developed under the Python programming language [6].
- SQLite stores the system and web commands in a local database.
- PyAudio detects hot words through audio capture [15].
- PyWhatKit is a Python library that allows videos to be played on YouTube [19].
- The Playsound function is used to play sound files.
- Web browser: Used to open Uniform Resource Locators (URLs) in the default web browser.
- Hugchat: a software tool that integrates a chatbot into conversations effortlessly [5].

B. *Functional Requirements:*

- For Command Execution: The Assistant must be competent with system commands and launching apps [3].
- For Media Playback: The Assistant is expected to start playback of videos on YouTube based on command [19].

The project overview is given in Figure 1 and Figure 2.

C. *Non-Functional Requirements:*

- Maintenance: An assistant must complete all tasks quickly and accurately per the predetermined set of standards. [8].
- Usability: A system must be easily operated and offer great user interaction. [7]
- Responsiveness: An assistant needs to respond quickly to all user commands and requests. [8]
- Spoken Responses: A user's queries to the Assistant should be attended to as one engages in a dialogue. [5].

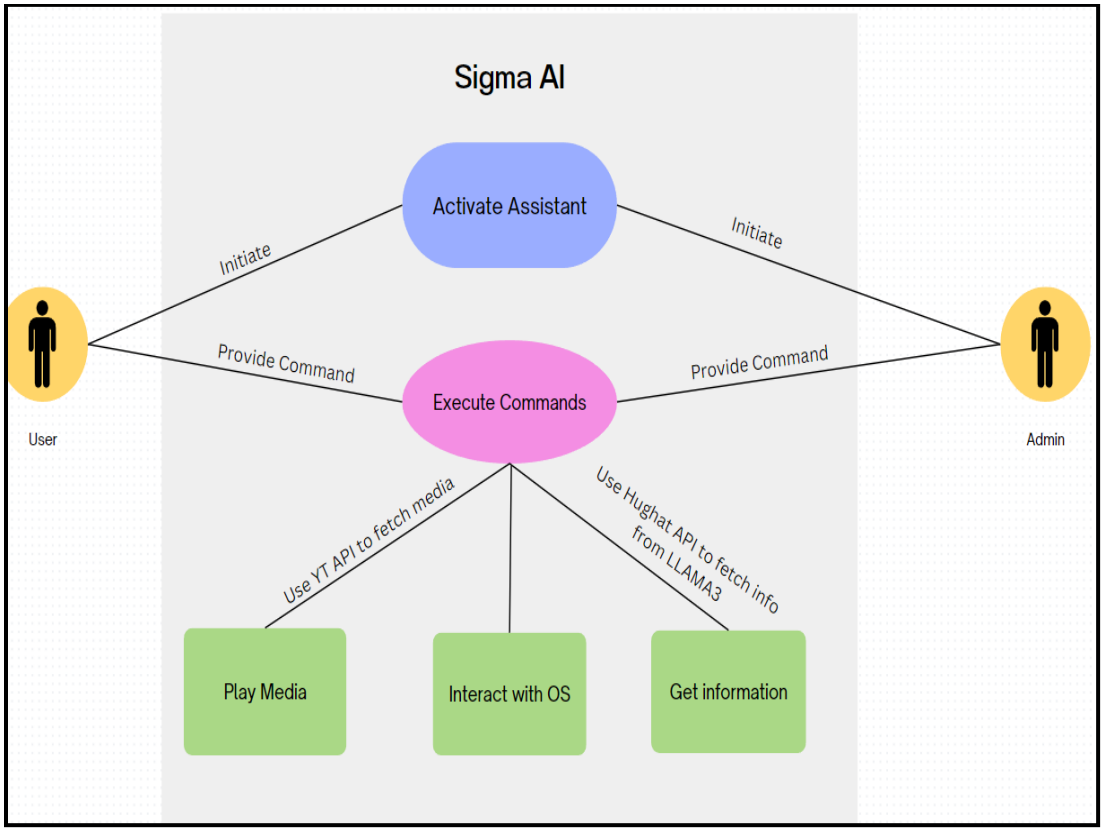

Figure 2 Use-case Diagram

Figure 3 Asking Interface

VII. RESULTS

Results are provided in the listed interfaces in Figures 3, 4, 5, 6, 7, and 8.

As presented in Figure 3, the user requests the name of the Assistant. Figure 4 shows that the Assistant is responding to the user. Figure 5 shows that the user asks the people who created this Assistant.

Figure 4 Responding Interface

Figure 5 Requesting Interface

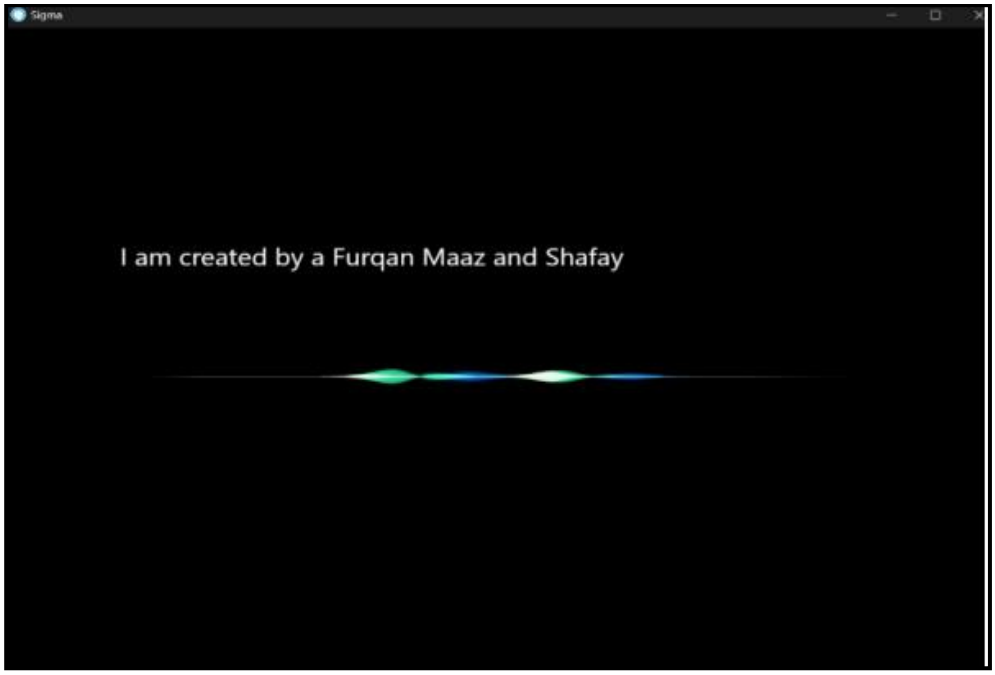

Figure 6 Responding Request Interface

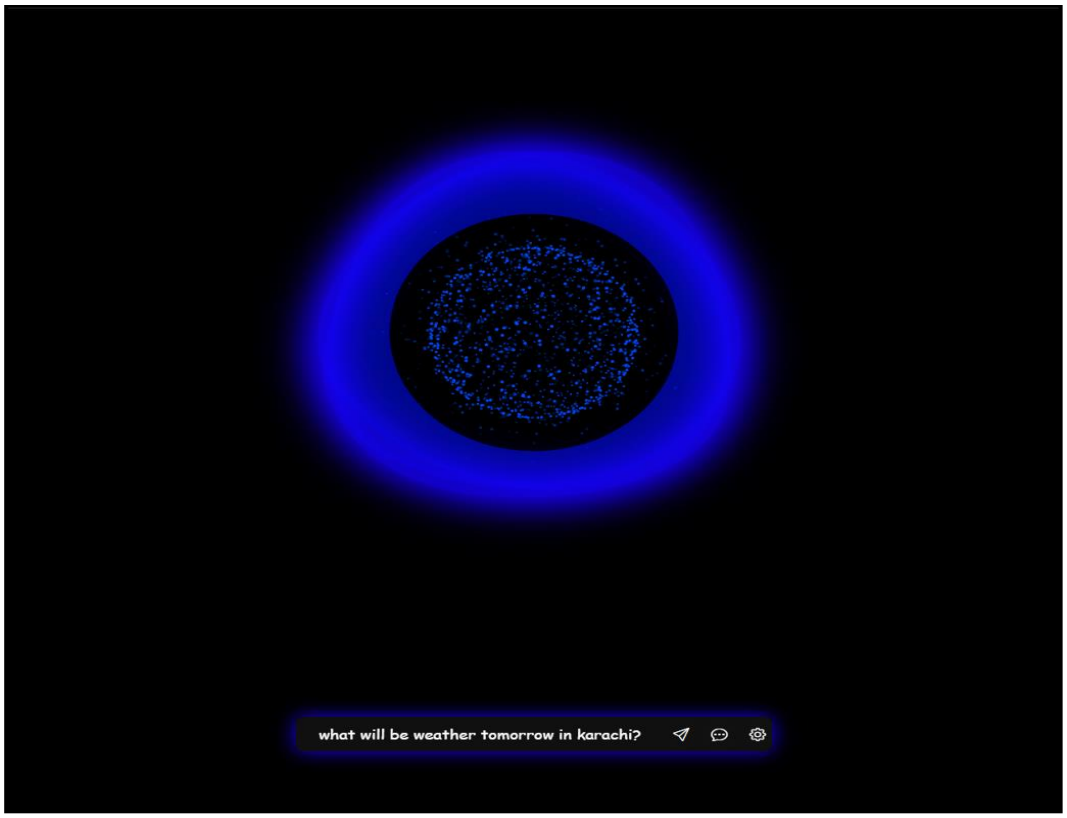

Figure 7 Requesting Interface with another query

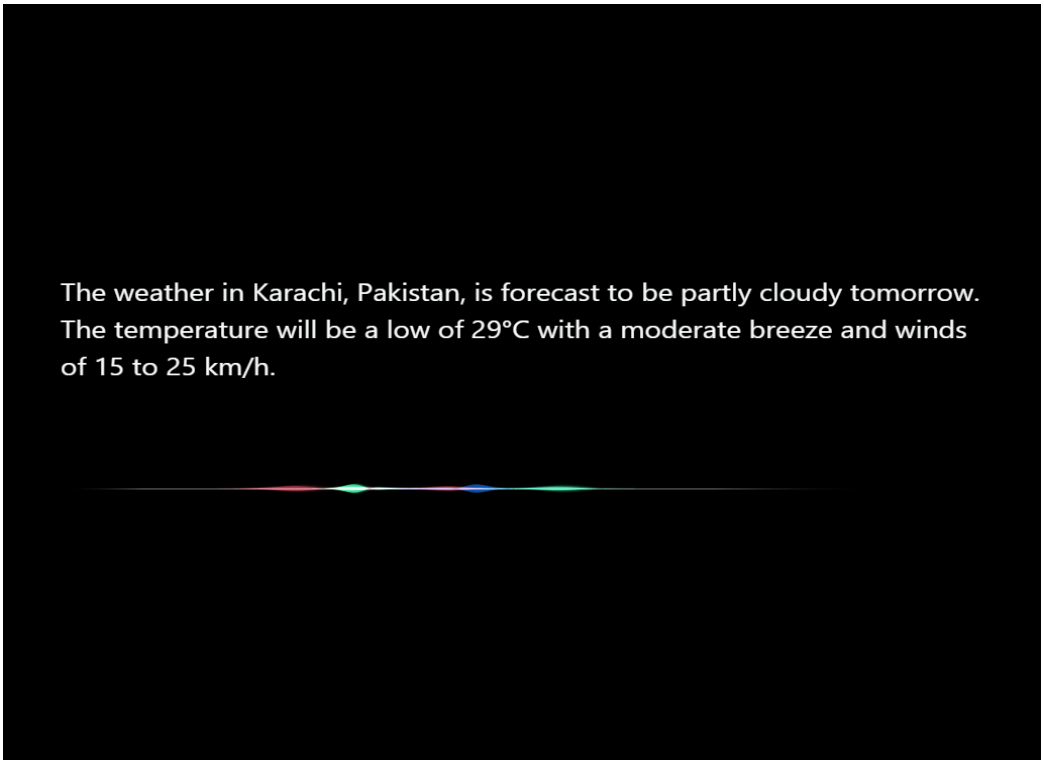

Figure 8 Responding Interface with Response

CONCLUSION

Developing the virtual assistant "Sigma" illustrates the capability of integrating sophisticated AI and machine learning technologies into a single, powerful and versatile digital assistant. This is achieved through hot word recognition, execution of system commands, conversational responses via chatbot, and playing videos on YouTube. This project embodies the application of natural language processing (NLP), machine learning (ML), and voice recognition technologies, enabling a user-friendly, smooth-flowing natural interaction to be processed with a machine. As discussed above, Sigma sets a pronunciation crown with other virtual assistants as they continue to advance in what can be achieved using modern-day tools and libraries in the Python programming language. The primary aim of Sigma, built with advanced AI technology, is to explore the potential of such technologies and redefine interaction boundaries between humans and virtual assistants. Advances in AI technology are encompassed in Sigma, and the evolution of virtual assistants is making AI technology redefine our digital interaction environments.

ACKNOWLEDGEMENTS

We deeply appreciate everyone who contributed to these ideas, changing them from a simplified version to one that incorporates actual reality. We understand the administrations from our different universities for their support and assistance in monitoring our research work. Family and friends offer critical support and understanding, which is fundamental in achieving our goals, irrespective of how difficult it is, as they nurture our happiness. We especially wish to thank the support staff who assisted us with materials, data gathering, and research and helped compile this paper. In the end, the study's authors acknowledge their colleagues' and co-authors' courtesy and appreciation for the great help and timely communication even when they were out of reach physically. This joint effort remarkably improved the distinctiveness and originality of the research.

REFERENCES

[1] Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. Neural Computation, 9(8), 1735-1780.

This paper introduces Long Short-Term Memory (LSTM), a recurrent neural network (RNN) architecture crucial for many natural language processing tasks.

[2] Vinyals, O., & Le, Q. (2015). A Neural Conversational Model. arXiv preprint arXiv:1506.05869.

Discusses a sequence-to-sequence neural network model for conversational agents.

[3] Cho, K., Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. arXiv preprint arXiv:1406.1078.

Explores the use of RNN encoder-decoder architecture, foundational for modern virtual assistants.

[4] Kumar, A., Gupta, P., & Sharma, S. (2018). Smart Virtual Assistant: A Review. Journal of Computational and Theoretical Nanoscience, 15(3), 791-796.

Provides a comprehensive review of smart virtual assistants and their underlying technologies.

[5] Chen, Q., Zhuo, Z., & Wang, W. (2019). Bert for Joint Intent Classification and Slot Filling. arXiv preprint arXiv:1902.10909.

Discusses BERT, a powerful NLP model for tasks like intent classification and slot filling, crucial for virtual assistants.

[6] Sarikaya, R., Hinton, G., & Deoras, A. (2014). Application of Deep Belief Networks for Natural Language Understanding. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 22(4), 778-784.

Examines the use of deep belief networks in natural language understanding.

[7] Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving Language Understanding by Generative Pre-Training. OpenAI.

Introduces the concept of generative pre-training for language models, a precursor to models like GPT-3.

[8] Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., ... & Dean, J. (2016). Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. arXiv preprint arXiv:1609.08144.

Discusses Google's neural machine translation system, which uses sequence-to-sequence models.

[9] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv preprint arXiv:1810.04805.

Introduces BERT, a model that has significantly improved various NLP tasks.

[10] Serban, I. V., Lowe, R., Charlin, L., & Pineau, J. (2015). A Survey of Available Corpora for Building Data-Driven Dialogue Systems. arXiv preprint arXiv:1512.05742.

Surveys the datasets available for training dialogue systems.

[11] Deng, L., & Liu, Y. (2018). Deep Learning in Natural Language Processing. Springer.

Comprehensive overview of deep learning techniques in NLP.

[12] Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to Sequence Learning with Neural Networks. Advances in Neural Information Processing Systems, 27, 3104-3112.

Fundamental paper on sequence-to-sequence models used in NLP.

[13] Young, T., Hazarika, D., Poria, S., & Cambria, E. (2018). Recent Trends in Deep Learning Based Natural Language Processing. IEEE Computational Intelligence Magazine, 13(3), 55-75.

Reviews recent trends in deep learning applied to NLP.

[14] Schmidhuber, J. (2015). Deep Learning in Neural Networks: An Overview. Neural Networks, 61, 85-117.

Provides an extensive overview of deep learning in neural networks.

[15] Amodei, D., Ananthanarayanan, S., Anubhai, R., Bai, J., Battenberg, E., Case, C., ... & Ng, A. Y. (2016). Deep Speech 2: End-to-End Speech Recognition in English and Mandarin. International Conference on Machine Learning (ICML).

Discusses end-to-end speech recognition systems.

[16] López, G., Quesada, L., & Guerrero, L. A. (2017). Alexa vs. Siri vs. Cortana vs. Google Assistant: A Comparison of Speech-Based Natural User Interfaces. International Conference on Applied Human Factors and Ergonomics. Springer, Cham.

Compares various virtual assistants and their speech-based interfaces.

[17] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... & Amodei, D. (2020). Language Models are Few-Shot Learners. arXiv preprint arXiv:2005.14165.

Discusses GPT-3, a state-of-the-art language model for various NLP tasks.

[18] Feng, S., Xing, L., Qian, Y., Chen, G., & Huang, C. (2021). Challenges in Reinforcement Learning for Conversational AI: A Survey. Journal of Artificial Intelligence Research, 71, 309-348.

Reviews the challenges and approaches in applying reinforcement learning to conversational AI.

[19] Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2017). Bag of Tricks for Efficient Text Classification. arXiv preprint arXiv:1607.01759.

Explores simple and efficient methods for text classification, which are essential for intent recognition in virtual assistants.

[20] Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. arXiv preprint arXiv:1408.5882. Discusses the application of convolutional neural networks (CNNs) for sentence classification tasks.