

AI Based Smart Robot (Chatbot) using Python

Bembrekar Ayan Ahamad¹; Solanke Vikas S.²; Ghadge Madhusudan Ramdas³
Department of Computer Engineering^{1,3}; HOD of Computer Department²
Marathwada Mitra Mandal's Polytechnic, Thergoan, Pune-411033

Abstract:- This research paper explores the development and functionality of the AI Based smart Robot, which includes an interactive chatbot. A chatbot, in its essence, is an artificially intelligence computer program that performs communication using the audio system, A chatbot, created with the Gemini API, is basically a talking robot that build using the Gemini API toolbox. This chatbot, it listens to your voice, understands what you're saying and give responds back to you. It develop to assistance to users. Additionally, the project's application in Customer Service, Virtual Assistant, Healthcare, Education, Smart Home Control.

Keywords:- Raspberry pi, Smart Robot, Chatbot, Gemini API, Python, Natural Language Processing, Artificial Intelligence.

I. INTRODUCTION

The AI Based, Smart Robot project represents the integration of intelligence and modern technology to provide users with spiritual knowledge support in a totally new, and easy-to-understand way. Created in collaboration with experts in the fields of Education, smart home control, virtual assistance, and healthcare. The main themes of this project revolve around the development and implementation of a voice chatbot with a 3D-printed robot face. The project explores the dynamics of interaction between humans and computers, focusing on voice-based communication and the integration of visual cues through the 3D-printed robot face. It aims to create a more natural and engaging user experience by combining voice commands with facial expressions.

The core of the project is the chatbot, which can engage in meaningful conversations with users on a wide variety of topics by acting as a virtual guide and advisor. By leveraging natural language processing algorithms and machine learning technology, Robot can understand users' questions, provide answers, and offer recommendations based on personal preferences and needs.

Creating a Smart Robot involves several key steps to ensure its effectiveness and usability. Firstly, it's essential to clearly define the purpose of the chatbot and understand the audience it will serve. Whether it's for customer service, virtual assistance, or another specific function, having a clear goal in mind helps guide the development process. Once the purpose is establish, selecting an appropriate platform or framework becomes crucial. Platforms like Google Dialog flow, Amazon Lex, or Microsoft Bot Framework offer tools and resources to streamline the development process, making

it easier to integrate speech recognition and natural language processing capabilities.

Next, designing the conversational flow is supremely important to creating a smooth user experience. This requires mapping out the dialogue between the user and the chatbot, including greetings, responses to regular queries, and error handling mechanisms. Designing a flowchart chat or using tools for conversational design could assist in visualizing the interaction and make sure that the chatbot responds appropriately to user input. Additionally, integrating speech recognition technology enables the chatbot to comprehend spoken commands or queries, while natural language processing allows it to interpret the meaning behind the user's words.

In addition to the chat function, the Robot chatbot also it integrates within the Gemini API, allowing it to access a lot more information from google and provide updates in real-time with client. The use of advanced technology and artificial intelligence in conjunction with the AI Based Smart Robot aims to typically revolve around truly enhancing user experiences, improving efficiency, Efficient Communication, personalized Assistance, Automation of tasks.

II. EASE OF USE

A. Navigating the Voice Chatbot Interface

To ensure a seamless user experience, navigating the voice chatbot interface should be like, you know, intuitive and straight forward, right? The users should totally be able to easily, like, access and interact with the chatbot using voice commands or maybe other input methods, you feel me? The interface design should, like, prioritize clarity and simplicity, duh, minimizing cognitive load and reducing, like, the need for extensive user guidance.

B. Providing Clear Instructions and Prompts

Clear and concise instructions are essential for guiding users through the chatbot interaction process. The chatbot should provide prompts and cues to help users understand what actions they can take and how to proceed. Instructions should be delivered in a natural and conversational manner, ensuring that users feel comfortable and confident in interacting with the chatbot.

C. Minimizing User Effort

Efficiency is key to enhancing the ease of use of a voice chatbot. The chatbot should be able to anticipate user needs and proactively provide relevant information or assistance. By minimizing the need for users to repeat themselves or

provide redundant information, the chatbot can streamline interactions and improve user satisfaction.

III. LITERATURE REVIEW

Numerous studies and advancements have been done in the field of artificial intelligence (AI), and as digital networks continue to grow quickly and become more widely used, the demand for timely and always-available services has intensified. One well-liked and rapidly growing method of service providers is the chatbot. Here are a few research studies that we have consulted in order to comprehend and develop our concept.

"Chatter Bot Algorithm" was developed by Michael Maudlin in 1994 and used to respond to queries. It was included in the book *Julia*. Based on this first concept, several initiatives to build a chatbot system were established. To utilize the Chat-Bot program, the user must log in. It is at that precise moment that the user may file complaints and inquiries. When a user submits a query to the bot, NLP is used after the query's context is identified. To discern the sentiment of the words, grammatical forms labeling and WordNet computation [4] are used. Questions from users are examined in the knowledge base. The user receives the answer if the correct response is found at that time. When a specific query cannot be answered by the database, the administrator will respond. The relevant response is given to the user at that precise moment, when the administrator responds to the question. Questions and answers are entered into a database so that, at a later time, these types of questions may be asked with the knowledge that the database will provide a valid response. As a result, the administrator is no longer required to physically address the same query. Suffixes in English words are removed using several techniques, such as the Porter Stemmer Algorithm [5]. To estimate the word request closeness between two sentences, utilize the word request vector approach. Sentences with the exact same words but a different word order might have very distinct meanings.

Researchers Ujjwal Kumar, Murai Jha, and Sonam Sirohi (May 2022) have studied the use of the Telegram platform for text-to-speech and text-based chatbots. It explains the steps involved in creating a Telegram bot. We talked about several cutting-edge subjects that we might tackle in the future, such text-to-speech conversion.

Surya Gunawan Teddy, Asaad Balla In January 2022, Falemlula Babiker, Nanang Ismail, and Mufid Ridlo Effendi conducted research and development to create a chatbot on Telegram that utilizes natural language processing (NLP).

In July 2020, Shreyashkar Sharma outlines the various stages involved in implementing a chatbot using the Python programming language. There are three ways to use the chatbot. The first technique is called pattern matching; it was created at the beginning of chatbot history and compares words entered by the user to determine the user's intention and produce pertinent results. The second approach is Natural Language Understanding (NLU), also known as natural

language processing, which takes user-sent speech or text and transforms it into data that a computer can understand.

In April 2020, Jaimit Dholakia provides a detailed explanation of his use of the Flask API and Python programming. His study served as a roadmap and source of inspiration for our project's use of the FLASK architecture. Flask API serves as a bridge between the various programs and applications that are utilized in our project.

In a June 2021 article, Kshitija Shingte, Anuja Chaudhari, Aditee Patil, Anushree Chaudhari, and Sharmishta Desai went into detail about how they created the chatbot with educational purposes in mind. Despite being created for the same profession—the educational field—this chatbot's design purpose differed slightly. It was an excellent resource for learning how to create a chatbot.

The study by Ramkumar Malvi and Ketan Dharurkaretal provides an overview of different approaches for implementing chatbots. The study's foundation is a survey of several chatbots, which demonstrates how they vary from one another. Various forms of technology have been utilized to create chatbots. A chatbot can be thought of as a knowledge-based question-answer system that uses experts to gather questions from users. A chatbot is a piece of software created to mimic a human-to-human dialogue. The purpose of this survey article is to provide a summary of current methods for putting chatbots into use. This paper presents a comparison of several chatbots, including the first chatbot, ELIZA, and the newest, ALEXA. Other chatbots that are included in the comparison are IBM Watson, Siri, Tay, and others. It provides information about their actual operation and how it is put into practice.

A thorough analysis of those systems is developed, along with the design and deployment of multiple chatbots. This paper discusses college administration. Chatbot The chatbots for college inquiries will be constructed with artificial algorithms that can comprehend and analyze user messages and requests. Through the chatbot, the user can ask questions about any college-related activity without having to visit the campus in person. After analyzing the query, the system provides the user with an answer. Any inquiry can be posed by the user. To enable this feature, the chatbot often remembers the prior command. Thus, the user will be remembered for everything they ask for, which will benefit the chatbot and enable analysis.

If the user's inquiry was not satisfactorily answered by the chatbot, the administrator will address it.

IV. PROJECT OVERVIEW

The project involves creating a voice-activated chatbot utilizing the Gemini API, which serves as a toolkit for developers. This chatbot enables users to interact with different services and applications through natural language voice commands. The primary focus lies in designing a seamless user experience while harnessing the capabilities of the Gemini API to facilitate voice-based interactions

effectively. Key components of the project include implementing speech recognition technology to comprehend user commands accurately, integrating natural language processing to interpret queries, and accessing necessary data and functionalities through the Gemini API.

A. Implementation

To implement a AI Based Smart Robot, we integrate a microphone with the Raspberry Pi to enable input through voice commands. This hardware addition allows the chatbot to listen to user queries and commands directly. The Robot needs to understand what users are saying, and for that, it requires speech recognition technology. This involves integrating software that can listen to spoken commands, convert them into text, and understand the meaning behind the words. This is crucial for enabling the chatbot to respond accurately to user inquiries and commands.

Natural language understanding (NLU) is essential for interpreting the meaning and context of user queries. NLU algorithms help the chatbot comprehend the intent behind user input, even if it's phrased in different ways. This ensures that the chatbot can provide relevant and helpful responses, regardless of how users phrase their requests.

The implementation of a voice chatbot also involves leveraging the Gemini API to access relevant data and functionalities. The Gemini API serves as a toolkit for developers, providing access to various resources and services that can enhance the capabilities of the chatbot. By integrating the Gemini API into the chatbot's backend infrastructure, developers can access data from external sources, such as weather forecasts, stock prices, or product information, to enrich the chatbot's responses and provide users with valuable insights and assistance.

Furthermore, an important aspect of the implementation involves adding a speaker to the Raspberry Pi for output. Once the chatbot processes user input and generates responses, these responses need to be conveyed back to the user in an audible format. By integrating a speaker with the Raspberry Pi, the chatbot can vocalize its responses, allowing users to hear the information or instructions provided.

Implementation involves creating a 3D robot face using CATIA software. This process begins with designing the physical appearance of the robot face, including features such as eyes, mouth, and facial expressions. In the implementation of the voice chatbot, another key component involves attaching a servo motor to the Raspberry Pi to control the movement of the robot's tongue.

B. Flowchart

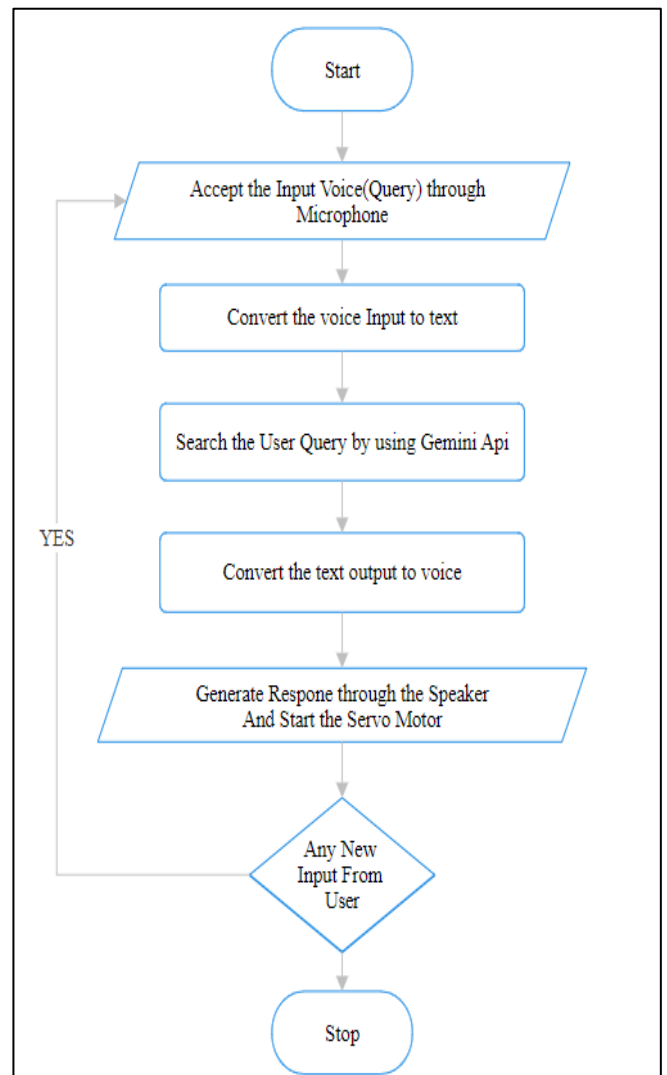


Fig. 1: Flowchart of AI based Smart Robot

C. Robot Face



Fig. 2: Robot Face

V. CONCLUSION

In the implementation of the AI Based Smart Robot, another key component involves attaching a servo motor to the Raspberry Pi to control the movement. In conclusion, the development of a voice chatbot utilizing the Gemini API, along with the integration of a servo motor to control the robot's tongue movement, represents a significant advancement in human-computer interaction. Through the implementation of speech recognition, natural language understanding, and access to external data sources via the Gemini API, the chatbot is equipped to understand user queries and provide relevant responses effectively. Additionally, the inclusion of a servo motor adds a physical dimension to the interaction, allowing the chatbot to simulate human-like movements and gestures, thereby enhancing the overall user experience. This combination of advanced technology and hardware integration demonstrates the potential for voice chatbots to become increasingly sophisticated and interactive, catering to a wide range of applications and user preferences of the robot's tongue.

REFERENCES

- [1]. A guide to Natural Language Processing, Available at https://en.wikipedia.org/wiki/Natural_language_processing
- [2]. Amey Tiwari, Rahul Talekar, Prof.S.M.Patil, "College Information Chat Bot System" International Journal of Engineering Research and General Science (IJERGS) Volume: 5, Issue: 2, Page no: 131-137| March-April 2017.
- [3]. E. Mahdiyah and Y. Andriyani, "Analisa Algoritma Pemahaman Kalimat Pada ALICE Chat Bot Dengan Menggunakan Artificial Intelligence Markup Language (AIML)" [2013].
- [4]. Jimit Dholakia, "Creating RESTful Web APIs using Flask and Python", Published in (Towards Data Science) Apr 26, 2020
- [5]. Shreyaskar Sharma, "Chatbot Development Using Python", Published in International Journal of Research Thoughts (IJCRT), July 7th, 2020.
- [6]. Kshitija Shingte, Anuja Chaudhari, Aditee Patil, Anushree Chaudhari, Sharmishta Desai, "Chatbot Development for Educational Institute", Published in SSRN, June 6th 2021.
- [7]. Ujjwal Kumar, Murai Jha, Sonam Sirohi, "Smart Telegram Chatbot", Published in Journal of Emerging Technologies and Innovative Research, vol. 9, May 2022.
- [8]. Teddy Surya Gunawan, Asaad Balla Falelmula Babiker, Nanang Ismail, Mufid Ridlo Effendi, "Development of Intelligent Telegram Chatbot Using Natural Language Processing", Published on IEEE, January 18th 2022.