Development and Implementation of Chatbot Using Machine Learning

A PROJECT REPORT

Submitted by,

CHAKSHU	-	**20211CSE0146**
CHOLESHWAR KUMAR	-	**20211CSE0006**
AMARJEET KUMAR	-	**20211CSE0007**
SAMRAT SARKAR	-	**20211CSE0098**
KOMMA BHANU PRAKASH REDDY	-	**20211CSE0108**

Under the guidance of,

Dr. Iqbal Gani Dar

Assistant professor, School of Engineering

Presidency University, Bengaluru

in partial fulfilment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

At

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

PRESIDENCY UNIVERSITY

BENGALURU

JANUARY 2025

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE ENGINEERING

CERTIFICATE

This is to certify that the Project report "**Development and Implementation of Chatbot Using Machine Learning"** being submitted by "**Chakshu, Choleshwar kumar, Amarjeet Kumar, Samrat Sarkar, Komma Bhanu Prakash Reddy**" bearing roll numbers "20211CSE0146, 20211CSE0006, 20211CSE0007, 20211CSE0098, 20211CSE0108" in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering is a bonafide work carried out under my supervision.

Dr.Iqbal Gani Dar
Assistant Professor
School of CSE&IS
Presidency University

DR. MOHAMMED ASIF H B
Professor & HoD
School of CSE&IS
Presidency University

Dr. L. SHAKKEERA
Associate Dean
School of CSE&IS
Presidency University

Dr. MYDHILI NAIR
Associate Dean
School of CSE&IS
Presidency University

Dr. SAMEERUDDIN KHAN
Pro-Vc School of Engineering
Dean -School of CSE&IS
Presidency University

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE ENGINEERING

DECLARATION

We hereby declare that the work, which is being presented in the project report entitled **Development and Implementation of Chatbot using Machine Learning** in partial fulfillment for the award of Degree of **Bachelor of Technology in Computer Science and Engineering**, is a record of our own investigations carried under the guidance of **Dr. Iqbal Gani Dar Assistant Professor, School of Computer Science Engineering & Information Science, Presidency University, Bengaluru.**

We have not submitted the matter presented in this report anywhere for the award of any other Degree.

Name	Roll No.	Signature
CHAKSHU	20211CSE0146	
CHOLESHWAR KUMAR	20211CSE0006	
AMARJEET KUMAR	20211CSE0007	
SAMRAT SARKAR	20211CSE0098	
KOMMA BHANU PRAKASH REDDY	20211CSE0108	

ABSTRACT

This project aims to develop a modern customer service chatbot that stands out with its unique "speak aloud" feature. Unlike traditional chatbots that rely solely on text communication, our chatbot uses machine learning to enhance customer engagement by efficiently responding to spoken inquiries. Along with text messaging, the chatbot includes several convenient features, such as a microphone for voice input, an image-to-text converter, and a quick copy option. This innovative approach is designed to deliver fast, simple, and flexible answers to customer questions, revolutionizing the way customer service is delivered with a well-structured and comprehensive solution.

ACKNOWLEDGEMENT

First of all, we indebted to the **GOD ALMIGHTY** for giving me an opportunity to excel in our efforts to complete this project on time.

We express our sincere thanks to our respected dean **Dr. Md. Sameeruddin Khan**, Pro-VC, School of Engineering and Dean, School of Computer Science Engineering & Information Science, Presidency University for getting us permission to undergo the project.

We express our heartfelt gratitude to our beloved Associate Deans **Dr. Shakkeera L and Dr. Mydhili Nair,** School of Computer Science Engineering & Information Science, Presidency University, and **Dr. Asif Mohammed**, Head of the Department, School of Computer Science Engineering & Information Science, Presidency University, for rendering timely help in completing this project successfully.

We are greatly indebted to our guide **Dr. Iqbal Gani Dar, Assistant Professor** School of Computer Science Engineering & Information Science, Presidency University for their inspirational guidance, and valuable suggestions and for providing us a chance to express our technical capabilities in every respect for the completion of the project work.

We would like to convey our gratitude and heartfelt thanks to the PIP2001 Capstone Project Coordinators **Dr. Sampath A K, Dr. Abdul Khadar A and Mr. Md Zia Ur Rahman,** Department Project Coordinators and Git hub coordinator **Mr. Muthuraj.**

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

Chakshu (1)

Choleshwar Kumar(2)

Amarjeet Kumar(3)

Samrat Sarkar(4)

Komma Bhanu Prakash Reddy(5)

LIST OF FIGURES

TABLE OF CONTENTS

CHAPTER-1
INTRODUCTION

1.1. Objective

The development of AI-powered chatbots is inspired by the need for traditional websites to offer a chat feature that allows bots to interact with users and address their queries efficiently. Unlike live agents, who are typically limited to handling two or three tasks at a time, chatbots can manage an unlimited number of interactions, significantly boosting operational efficiency. For organizations such as schools or businesses dealing with a high volume of inquiries, chatbots can help lighten the workload of the support team. Furthermore, chatbots are highly effective at improving response times compared to human agents. An additional advantage is that millennials tend to favor live chat over phone calls, making chatbots an ideal solution to meet this preference.

Chatbots offer a dynamic and highly engaging platform for interactive marketing. They are particularly effective at automating repetitive tasks, which can save time and effort for businesses and educational institutions. For example, when a school or company frequently receives the same types of inquiries, the support team must respond to each question repeatedly. One of the biggest advantages of chatbots is their 24/7 availability, allowing users to resolve their queries at any time without delays. These benefits highlight the motivation for implementing a college inquiry chatbot. Several well-known chatbots, such as Amazon Alexa, Google Assistant, Siri, and Bixby, demonstrate the value of such technology. For instance, consider the Amazon shopping app. Customers might face challenges finding information on how to return a product. In the past, they would have to call customer service and endure long wait times to speak with a representative—a frustrating and time-consuming experience. To simplify this process, Amazon introduced a chatbot that can efficiently handle simple customer requests, making the entire experience much more user-friendly.

1.2. Problem Statement

Creating a chatbot that can engage in human-like conversations while providing accurate and context-aware responses is an exciting but challenging task for both businesses and developers. Chatbots are becoming increasingly popular across various sectors, including customer service, education, healthcare, and entertainment. They play diverse roles, such as virtual assistants, educators, or even companions.

However, designing a chatbot that truly understands user intent, maintains context throughout a conversation, and delivers timely, relevant responses is no simple task. Achieving this requires a combination of advanced language processing, adaptability to user needs, and a natural conversational flow to ensure interactions feel seamless, authentic, and genuinely helpful.

1.3. Project Information

Chatbots are a form of human-computer interaction system that operates using natural language processing, either through text or speech. These automated systems typically function around the clock, making them highly efficient. Primarily designed to drive user engagement and handle a vast number of requests per hour, chatbots are capable of understanding commands, processing input, and completing tasks. Although chatbots have existed for quite some time, advancements in data availability, enhanced processing power, and open-source development frameworks have significantly improved their functionality. These advancements have led to the widespread adoption of chatbots across various industries and applications. Today, chatbots can be found in diverse settings, including customer support, social media platforms, online retail websites, and even banking services. Each chatbot is typically developed with specific tasks in mind. For example, customer service chatbots are specifically designed to assist and support customers in addressing their needs.

CHAPTER-2

LITERATURE SURVEY

[1] Akash Balachandar and Anusha D Kulkarni "Recruiments Chatbots",vol. 5, 08 Aug 2018

In this paper, the authors discuss the design of chatbots that act as human conversational partners, capable of understanding and interpreting meaningful human responses. In today's fast-paced world, gathering accurate information, especially during the hiring process, can be a challenging task. Implementing a chatbot offers a practical solution to this issue.

[2]Dr. Kevin Curran and Dr. Daniel Kelly, "Task-Based Interaction Chatbot:

User communication with the chatbot takes place on the front end, where Natural Language Understanding (NLU) plays a key role in interpreting the context and intent behind the user's input. Based on this understanding, the chatbot generates a relevant response. The chatbot's knowledge base, which is built through NLU, is maintained and supported on the back end.

[3] Harshala Gawade, Prachi Vishe, Vedika Patil, Sonali Kolpe, "Update the data as per the customer revier",2018.

They created a chatbot that leverages a knowledge database to generate responses. The system they proposed includes both an online inquiry feature and a chatbot service. The development process utilizes a range of programming languages to build intuitive graphical interfaces, making it easy for users to send and receive responses. Additionally, SQL is used for pattern matching to improve the chatbot's accuracy and functionality.

[4] Ms.Ch. Lavanya Susanna, R. Pratyusha, P. Swathi, P. Rishi Krishna, V. Sai Pradeep, "A review on customer support Chatbot Using Machine Learning Methods", 2021:

They developed a rule-based chatbot that offers users a predefined set of categories or questions to choose from, with answers provided only for the selected queries.

[5] Hrushikesh Koundinya K, Ajay Krishna Palakurthi, Vaishnavi Putnala, Dr. Ashok Kumar K "A survey on deep learning-based Chatbot " ,2022:

They designed a chatbot using Machine Learning (ML) and Python. It functions as a rule-based system: if a user's query matches the database, the chatbot provides a relevant response. If not, it delivers a predefined response.

[6] Gandhar Khandagale, Meghana Wagh, Pranali Patil, Prof. Satish Kuchiwale ,2023:

The chatbot presents a list of options to the user, who then selects the option number they want to inquire about. Upon the user's request, the chatbot provides a link to the relevant college or institution.

[7] Ravi R, Gautama Sinha"Intelligent Chatbot for Easy Web-Analytics Insights". In 2018 International Conference on Advances in Computing, Communications and Informatics

This paper presents a comparison based on the ease of use, utilizing various analytic tools. The chatbot is developed using Artificial Intelligence Markup Language (AIML), which contains raw analytics data. The necessary information is retrieved from the raw data provided by the analytics tools. Every website tracks the details entered by users.

CHAPTER-3

RESEARCH GAPS OF EXISTING METHODS

3.1. Natural Language Understanding (NLU) and Context Awareness

- **GAP:** Many chatbots face challenges in keeping track of context during extended conversations or seamlessly transitioning between different topics.
- **Opportunities:**
    - Create methods to improve memory retention and enable better adaptation to context.
    - Enhance the ability to manage multi-turn conversations with smooth transitions between changing topics.

3.1.1. Emotional Intelligence and Personalization

- **GAP:** They have limited emotional intelligence, making it difficult for them to accurately recognize and respond to the emotions of users.
- **Opportunities:**
    - Incorporate emotion-detection models and sentiment analysis to enable the chatbot to provide more empathetic responses.

3.1.2. Bias and Ethical Concerns

- **GAP:**Many chatbot systems unintentionally exhibit societal or algorithmic biases.
- **Opportunities:**
    - Improve datasets and training processes to minimize biases.
    - Develop transparency mechanisms to explain decision-making processes.

3.2. Language and Cultural Adaptability

- **GAP:** Insufficient support for diverse languages, dialects, and cultural nuances.
- **Opportunities:**
    - Expand language support, including underrepresented or regional languages.
    - Integrate cultural context to improve conversational relevance.

3.2.1. Domain-Specific Expertise

- **GAP:** Lack of specialization in domain-specific knowledge, resulting in generic or incorrect answers.
- **Opportunities:**
    - Develop domain-specific chatbots with robust knowledge graphs.
    - Use transfer learning to adapt general-purpose models to specific industries.

3.2.2. Evaluation Metrics

- **GAP:** Lack of standardized, comprehensive metrics to evaluate chatbot performance beyond accuracy and response time.
- **Opportunities:**
    - Develop holistic evaluation frameworks that include user satisfaction, engagement, and ethical considerations.

3.3. Scalability and Resource Efficiency

- **GAP:** High computational costs and resource inefficiency in training and deploying large-scale models.
- **Opportunities:**
    - Research energy-efficient model architectures and compression techniques.

- Optimize inference speeds for real-time interactions on edge devices.

3.3.1. Security and Privacy

- **GAP:** Vulnerabilities in data security and privacy protection during conversations.
- **Opportunities:**
  - Develop robust encryption and anonymization techniques.
  - Enhance transparency regarding data usage and user consent.

3.3.2. Multimodal Interaction

- **GAP:** Limited integration of text with other modalities (e.g., voice, image, video).
- **Opportunities:**
  - Expand chatbot capabilities to include multimodal inputs and outputs.
  - Research seamless transitions between text, voice, and visual media.

3.3.3. Long-Term Engagement

- **GAP:** Inability to maintain user engagement over time.
- **Opportunities:**
  - Design conversational strategies to sustain user interest and encourage continued interaction.
  - Incorporate gamification and dynamic content updates.

CHAPTER-4
PROPOSED SYSTEM

4. System Architecture

In this section, we are talking about System Architecture in detail in which we are discussing how user can interact with chatbot each and every step. First of all, user input the query. It will compile it in Natural Language programming and after compiling it send the data to Content Tracking and Dialogue Manager. Dialogue Manager check the sentence is written is meaningful or not if not it will show error. On the same time content tracking is tracing the sentence to check if the title is already present it takes data directly from the database otherwise it will train the data.

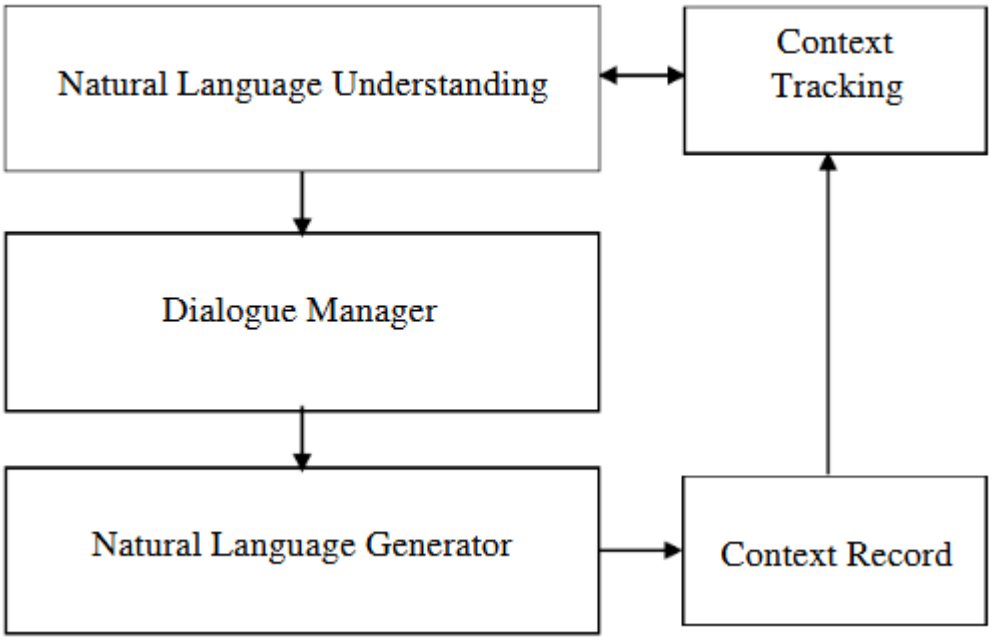


Figure 1. System Architecture

4.1. **Natural Language Understanding:** The first core component is Natural Language understanding in this step it checks the Topic Detection, Intent Analysis and Entity Linking. First we are talking about Topic Detection

in this topic can be selected and check if the topic should already present or not. If it is already present, then it shows error otherwise its correct. In the second step Intent analysis it checks every word is connected to other words/sentence or not.

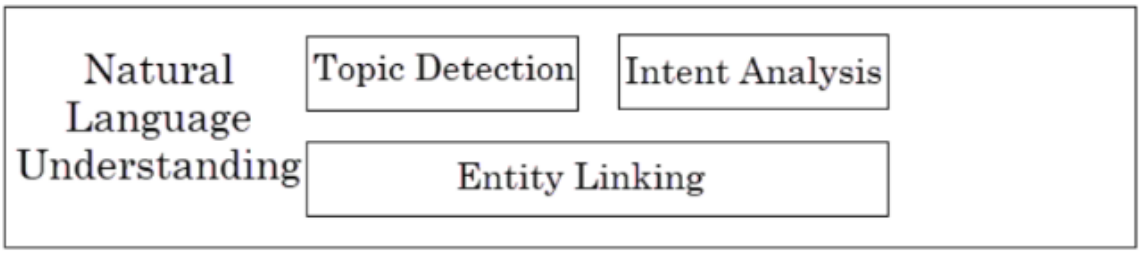


Figure 2: Natural Language Understanding [First Core Component]

4.2. **Context Tracking:** When a user submits a sentence, it checks the sentence and check the content is already present or not in which if the content is already present it shows error otherwise it passes to the next step.

4.3. **Dialogue Manager:** The Dialogue Manager (DM) is another essential component of any chatbot, it is of two types Rule based and Knowledge based. In Figure 3 shows that if Rule Based is Yes then everything is correct and its ready to move for next step, if No it is move to knowledge based. Then again checks in the knowledge base process if yes it will move to next step otherwise it moved to Retrieval Based and Generative Based.

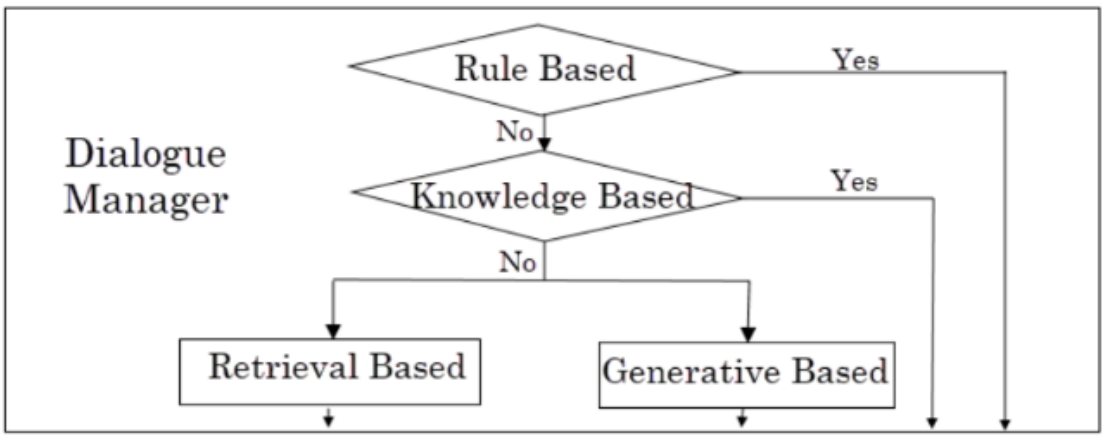


Figure 3: Dialogue Manager [Second Core Component]

4.4. **Natural Language Generator:** The last major core component of any chatbot. It is of two types Content Filter and Engagement Ranking. In content filter Natural Language Generator checks the content of the project is the content is already taken or present it reject the project. Give a pop up message to its already present choose new one. And the second step is Engagement Ranking if the article/report is passed to next step it uploaded to database and it checks the similarity and Ranking.

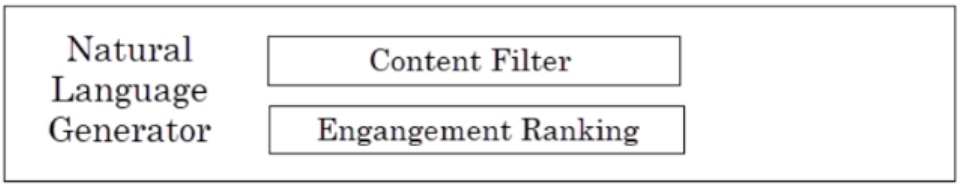


Figure 4: Natural Language Generator [Last Core Component]

CHAPTER-5

OBJECTIVES

5.1. Enhance User Experience

- Provide accurate, quick, and natural responses to user queries.
- Improve accessibility by enabling 24/7 availability for assistance.
- Support multilingual and multimodal interactions (text, voice, images).

5.2. Automate Repetitive Tasks

- Reduce human workload by automating FAQs, customer support, or routine inquiries.
- Streamline business processes like booking, order tracking, or troubleshooting.
- Minimize response time for standard queries.

5.3. Improve Contextual Understanding

- Leverage ML to understand user intents and extract relevant information accurately.
- Maintain conversation context over multi-turn dialogues for seamless interactions.
- Handle ambiguities, incomplete sentences, or corrections effectively.

5.4. Personalize User Interactions

- Use ML models to analyze user preferences, history, and behaviors.
- Provide tailored recommendations, suggestions, or responses.
- Adapt responses to match the tone and sentiment of the user.

5.5. Support Scalability

- Handle a large volume of simultaneous interactions without degrading performance.
- Dynamically adapt to different domains or tasks with minimal retraining.

5.6. Learn and Improve Over Time

- Use user interactions as feedback to train and refine ML models.
- Identify and address gaps in understanding or performance.
- Adapt to changing user behavior or evolving domain requirements.

5.7. Enable Multidomain Expertise

- Integrate knowledge across various fields (e.g., healthcare, finance, e-commerce).
- Provide accurate domain-specific information and services.
- Transition smoothly between domains during conversations.

5.8. Ensure Security and Privacy

- Safeguard user data and ensure compliance with privacy regulations.
- Use ML to detect anomalies and prevent fraudulent activities in interactions.
- Provide transparency regarding data usage and security measures.

5.9. Foster Long-Term Engagement

- Design dynamic and engaging conversational strategies to retain users.
- Incorporate interactive features like gamification or storytelling.
- Build trust and loyalty by ensuring consistent and empathetic interactions.

The goals focus on developing chatbots that go beyond simple automation, transforming them into intelligent and user-centric systems designed to meet a wide range of requirements. Would you like to delve deeper into particular objectives or explore specific use cases?

CHAPTER-6
SYSTEM DESIGN & IMPLEMENTATION

6.1. Introduction

Recent advancements in machine learning have enabled the development of significantly more intelligent chatbots. It provides automated support and services. Using Natural Language Processing and Machine Learning algorithm. It provides 24/7 accessibility and personalization. This chatbot provide customer service, user interaction. If the user requires any information which is not present in database, then its automatically train the data as per the user required. It takes less time to solve the problem, query and any other questions. In this chatbot we collected the user information and train the data. It interacts to customer through text or voice. It can respond to user queries instantly, reducing wait time.

6.2. Architecture

Figure 5. shows that Chatbot Development flow chart. In this flow chart User input the query and it will pass through different steps to generate output. From beginnings when the user giving a query as input like Text. After that the text will moved forwarded to Convert text as Lowercase letter. Next step is tokenization process in which all the text should be divided in sub-words and send to next step. In this step the program should check if the sentence or word are meaningful then if generate output for the next step. The next step is stemming it is the process of checking the prefix, suffix or infix. This check the grammatical uses in sentence is correct or not. If the grammatical is not in proper form, then it shows error. After all step's passed the program produce output and again ask Any New Input or End the task. It will help users to ask the query and solve the problem. It is very fast to find the solution, if the

solution is not in the database it automatically updates the data. It shows the data in easiest way like tabular form, numbering, using pointers.

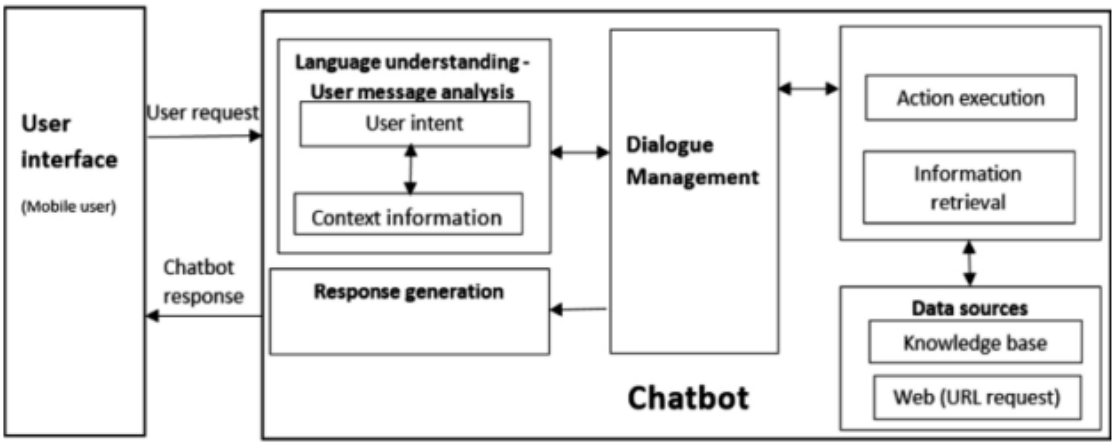


Figure 5: Chatbot Architecture

6.3. Natural Language Understanding

1. When the system gets a user request, the NLU component is employed to retrieve input information. The meaning that is subsequently produced by this information can then be applied in the process. Slot filling, purpose classification, and dialogue act classification are the three types of tasks that NLU completes.
2. **Classifying dialogue** acts is the process of interpreting user input. More specifically, it links one of the potential dialogue act kinds to the user's words. The user statement may take the form of an offer, statement, query, or other interaction act. Understanding the dialogue act helps us to better grasp the user's request and determine which response to provide.
3. **Intent classification** identifies the user's goal or action. These intents typically vary based on the subject matter. For example, a user's request might relate to ordering food, booking a hotel, checking the weather, and so on. In the context of hotel reservations, the user might want to book, cancel, or modify a reservation. Similarly, in the food ordering domain, the user could be looking to place, inquire about, or alter an order.

3.1. **Regular expressions** are a method for defining patterns in text. In these expressions, characters can either be regular characters or special meta-characters that serve specific functions.

3.2. **Tokenization** This is the process in which main sentence is substituted to sub sentence. All the words should be counted as Tokens. The process check if the sentence is meaningful or not. If the sentence is not meaningful then the sentence should be void. The text input which is collected from user and convert it to lower case letter. This avoid number value.

3.3. **Text normalization** refers to converting text into a consistent format. Word normalization involves identifying words with similar meanings but different spellings, such as "U.S.A." and "USA." Case folding is the process of converting all text into lowercase letters. Lemmatization focuses on identifying the root form of words and transforming their various forms into the base form, for instance, changing "produce," "product," "produces," and "production" to "produce."

3.4. **Bag of words,** The simplest way to process input is by using a bag of words, also known as the vector space model. This approach involves counting the frequency of each word, without considering grammar or word order. It can be implemented through text cleaning and removing stop words—words that don't contribute significant meaning to the sentence. However, the bag-of-words method has limitations, especially when the meaning of the text relies on language rules such as grammar or word order. For example, "Alice is taller than Bob" and "Bob is taller than Alice" would appear identical in the bag-of-words method, even though their meanings are different.

3.5. Latent Semantic Analysis(LSA) or Latent Semantic Indexing does not rely on any explicit understanding of language. Instead, LSA focuses on the meaning of words rather than the words themselves, grouping words that frequently appear together in similar contexts. To compare words, we create a matrix where each row represents a word, and each column represents a document. Each cell in the matrix indicates the frequency of a word's occurrence in a specific document. We then use Singular Value Decomposition to transform the matrix and reduce its complexity.

6.2. What is natural language processing?

Natural Language Processing (NLP) is a specialized branch of computer science that falls under artificial intelligence. Its primary goal is to equip computers with the ability to comprehend and engage with human language, both in written and spoken forms, in a manner similar to how humans do. NLP integrates rule-based linguistic models, derived from computational linguistics, with statistical approaches, machine learning, and deep learning techniques. These methods enable computers to interpret language data, grasp its full context, and infer both the intent and emotions behind it.

Natural Language Processing (NLP) drives numerous applications, such as translating text between languages, interpreting spoken commands, and generating concise summaries of extensive text—even in real-time scenarios. You've probably interacted with NLP through tools like voice-guided GPS systems, virtual assistants, speech recognition software, customer service bots, and other common conveniences.

NLP is becoming increasingly important in enterprise solutions beyond just consumer applications and optimizes essential business processes for organizations. By making interactions between humans and machines more natural, NLP is reshaping how we use technology in both personal and professional settings.

Figure 6: Diagram of Natural Language Processing

6.3. NLP tasks:

Human language is naturally intricate and filled with ambiguities, making it a significant challenge to create software that can interpret text or voice data with comparable accuracy. Elements like homonyms, homophones, sarcasm, idiomatic expressions, metaphors, grammatical exceptions, and diverse sentence structures add to the complexity of language. While it takes humans years to fully grasp these nuances, natural language processing applications must be designed to understand them from the beginning to function effectively.

Numerous NLP tasks assist in analyzing human text and voice data, allowing computers to understand and process the information effectively. Some of the essential tasks include:

- **Speech Recognition (Speech-to-Text):** This process transforms spoken language into written text. It is essential for applications that rely on voice

commands or provide verbal responses. Speech recognition is challenging due to the natural way people speak—rapidly, with slurred words, varying tones, diverse accents, and frequent grammatical errors.

- **Part-of-Speech Tagging (Grammatical Tagging** This involves identifying the grammatical roles of words based on their usage.
- **Word Sense Disambiguation:** Word Sense Disambiguation involves identifying the correct meanings of words with multiple definitions based on their context. For instance, it distinguishes between the verb "make" in the phrase "make the grade" (meaning "succeed") and "make a bet" (meaning "wager").
- **Sentiment Analysis:** Sentiment analysis involves extracting subjective elements from text, such as emotions, attitudes, sarcasm, confusion, or suspicion, that accompany the intended meaning, tone, and purpose of the words. Therefore, this analysis is a form of semantic analysis.
- **Natural Language Generation (NLG):** Natural Language Generation (NLG) is typically viewed as the counterpart to speech recognition. It involves converting structured data into clear, comprehensible human language, enabling machines to "speak" or generate meaningful, natural text.

6.4: NLP tools and approaches

Python offers a variety of tools and libraries for Natural Language Processing (NLP), with NLTK being a particularly useful open-source collection of libraries, programs, and educational resources for creating NLP applications. NLTK covers a broad spectrum of NLP tasks, including those listed above, and also provides additional libraries for specific sub-tasks such as plagiarism detection by paraphrasing sentences.

Besides this, the NLTK also has tools for understanding meaning, which help machines make logical conclusions from facts taken from text. This ability leads to a better understanding and smarter replies to human language.

Its comprehensiveness and extensibility make NLTK a fundamental resource for developers and researchers seeking to "humanize" their applications in ways that make more meaningful interactions with text and language data possible.

6.3.2. NLP CASE :

Natural NLP is the backbone of most AI systems in many current and real-world applications. Some of the significant examples include:

- **Spam Detection:**
  While often overlooked as an NLP solution, spam detection relies heavily on text classification capabilities. Advanced spam filters scan emails for language patterns indicative of spam or phishing. These include:
  - Overuse of financial terms.
  - Common grammatical errors.
  - Threatening or manipulative language.
  - Misspelled company names.
  - Inappropriate urgency.

    Spam detection is one of the few NLP challenges considered

"mostly solved," though some users might disagree based on their email experiences.

- **Social Media Sentiment Analysis:**

  Businesses are increasingly using NLP to extract insights from social media platforms. Sentiment analysis analyzes the language used in posts, comments, reviews, and other online interactions to determine attitudes, emotions, and opinions. Companies use this information to improve product designs, optimize marketing campaigns, and respond appropriately to customer feedback.

- **Text Summarization:**

  Text summarization uses Natural Language to condense large volumes of text into concise summaries or synopses. Applications include:

  - Creating abstracts for research papers and indexes.
  - Generating executive summaries for busy professionals.

    It utilizes advanced semantic reasoning coupled with NLG in an endeavor to offer valuable context along with insightful inferences in summaries.

These use cases indicate how NLP continues to "humanize" technology by allowing systems to interpret, analyze, and interact with human language effectively.

6.3.3. NEURAL NETWORK DIAGRAM:

This deep learning method gets its moniker because it functions similarly to how brain neurons process information. Its primary purpose is to identify trends between a dataset's input attributes and the related outputs.

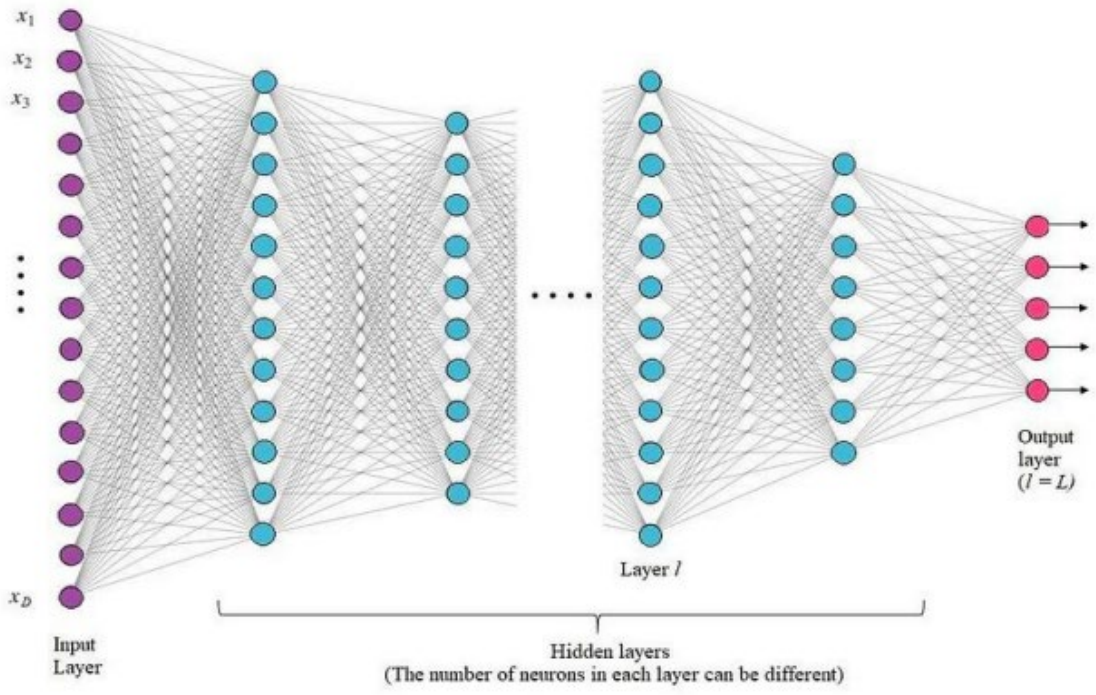


Figure 7 : Neural Network Architecture

Purple circles in the above graphic represent and are merely data set features. Hidden layer neurons are shown by the blue circles. These layers acquire the mathematical knowledge needed to correlate the inputs and produce the outputs. Lastly, we have the output layer's pink circles, which are represented by 13. It is equivalent to how many distinct classes are used. As an illustration, consider a 5x4 dataset with values for 4 features in 5 input vectors.
Assume that each row should be categorized as either good or bad, with 0 denoting good and 1 denoting terrible.

1. **Forward Pass through a Feed-Forward Neural Network**

   This stage uses a number of hidden layers to connect the input and output layers. The weighted sum of the input vector elements (xi) and the bias term b are received by the first layer of neurons (l=1). The output z is then produced by each neuron applying a differentiable nonlinear activation function h(•) to the weighted sum it receives, represented by the symbol a.

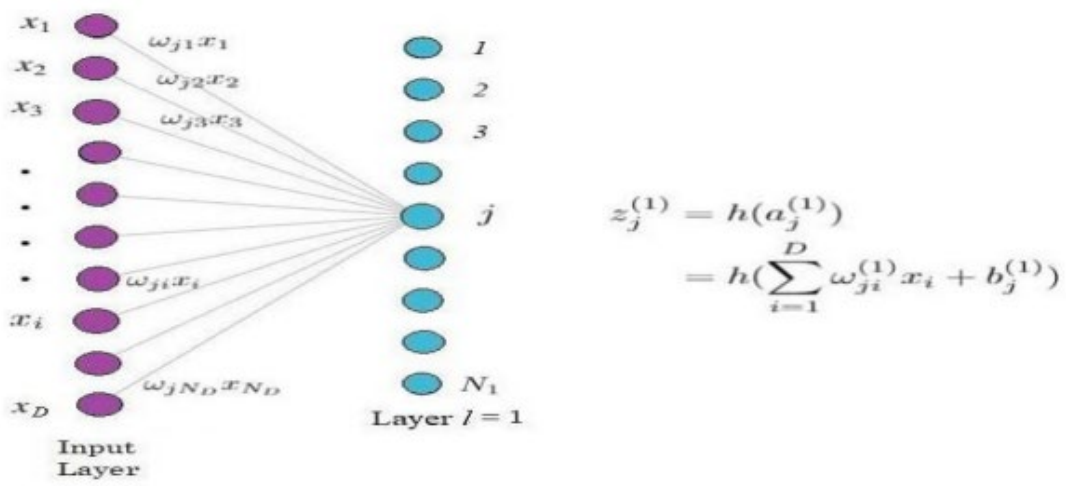


Figure 8 : Hidden Layers of Neural network Architecture

The procedure keeps going until the final layer's neurons' outputs (l = L) are evaluated. The output layer's neurons are essential for identifying the class to which the input vector belongs. The class linked to the neuron with the highest output value is given input vectors. Layer to layer, the activation function may change. The Rectified Linear Unit (ReLU) function and the SoftMax function are the two activation functions that we use the most for our chatbots. Usually, the output layer uses the SoftMax function, and the hidden layer applies the ReLU function.

Because it produces a probabilistic output, the SoftMax function is favored in the output layer. If the input is positive, the Rectified Linear Unit (ReLU) function, which is a piecewise linear function, outputs the input directly; if not, it outputs zero. The ReLU function has the following definition:

$$f(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases}$$

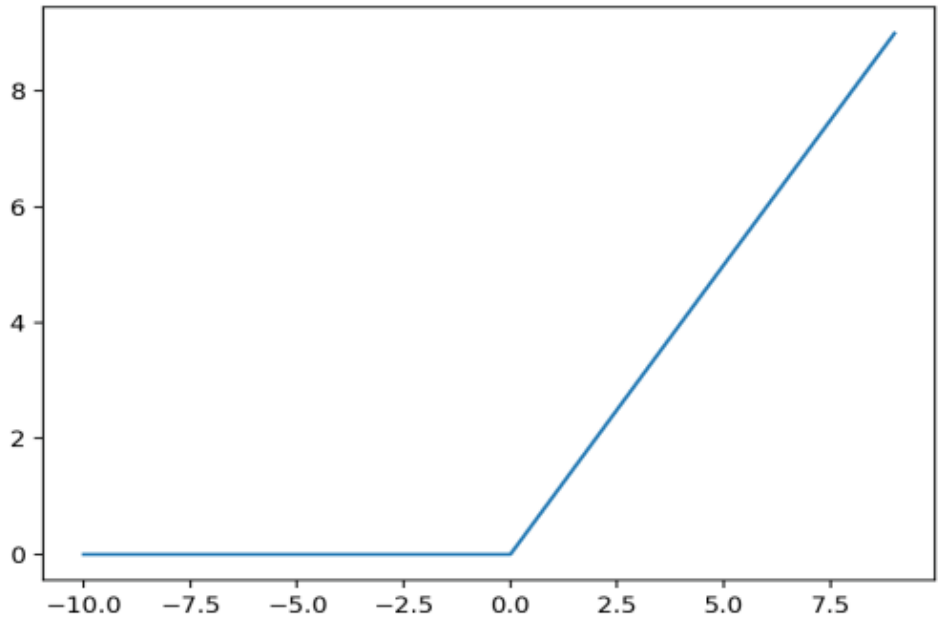


Figure 9: ReLU Function

The output layer of neural network models that forecast a multinomial probability distribution frequently uses this function as the activation function. Stated differently, softmax functions as the activation function for multi-class classification problems that require class membership determination among more than two labels.

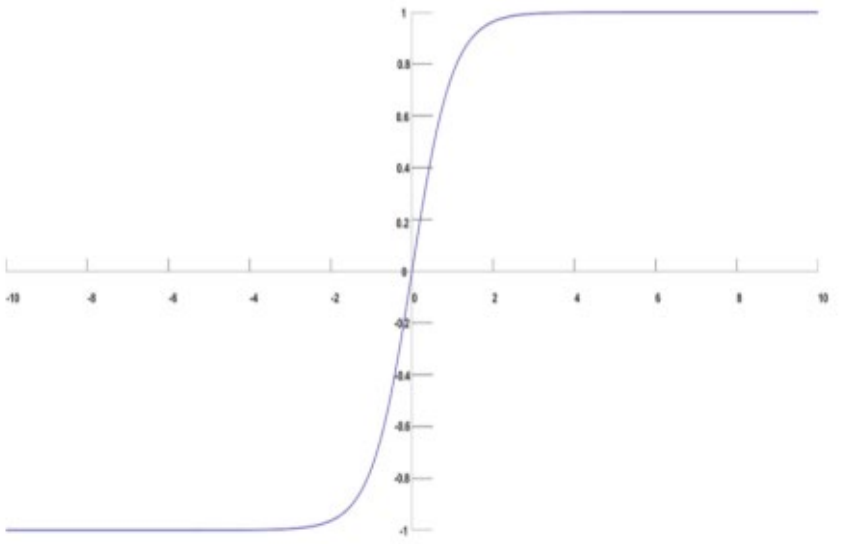


Figure 10 : Softmax Activation Function

2. **Backpropagation of Error to train Neural Network**

   This is an important step. Here, a neural network algorithm's job is to figure out which weights are appropriate for each layer so that the output is correct. In essence, we are attempting to identify the 16 suitable biases and weights. Consider an input vector that is known to belong to class A and is supplied into the network. Our forecast is wrong if the output layer shows that class B has the greatest value. To properly update the weights and biases, we must backpropagate the error as we can only compute it at the output.

   BAG-OF-WORDS (BOW) MODEL: Before using any neural network algorithm, text must be converted into numerical form because computers can only process numbers. The Bag-of-Words (BoW) model, which is frequently used to represent text as numbers and makes machine learning algorithms possible, is one efficient way to do this.Think around five sentences to get this.

6.5. CLASS DIAGRAM

One kind of static diagram that shows the structure of an application is a class diagram. It helps with the development of executable code for software applications and can be used to visualize, describe, and document different parts of a system.
This diagram shows the characteristics and functions of a class as well as the system-wide limitations. Because class diagrams may be readily transferred to object-oriented programming languages, they are very useful for designing object-oriented systems.
A class diagram, often known as a structural diagram, essentially shows a group of classes, interfaces, affiliations, collaborations, and constraints.

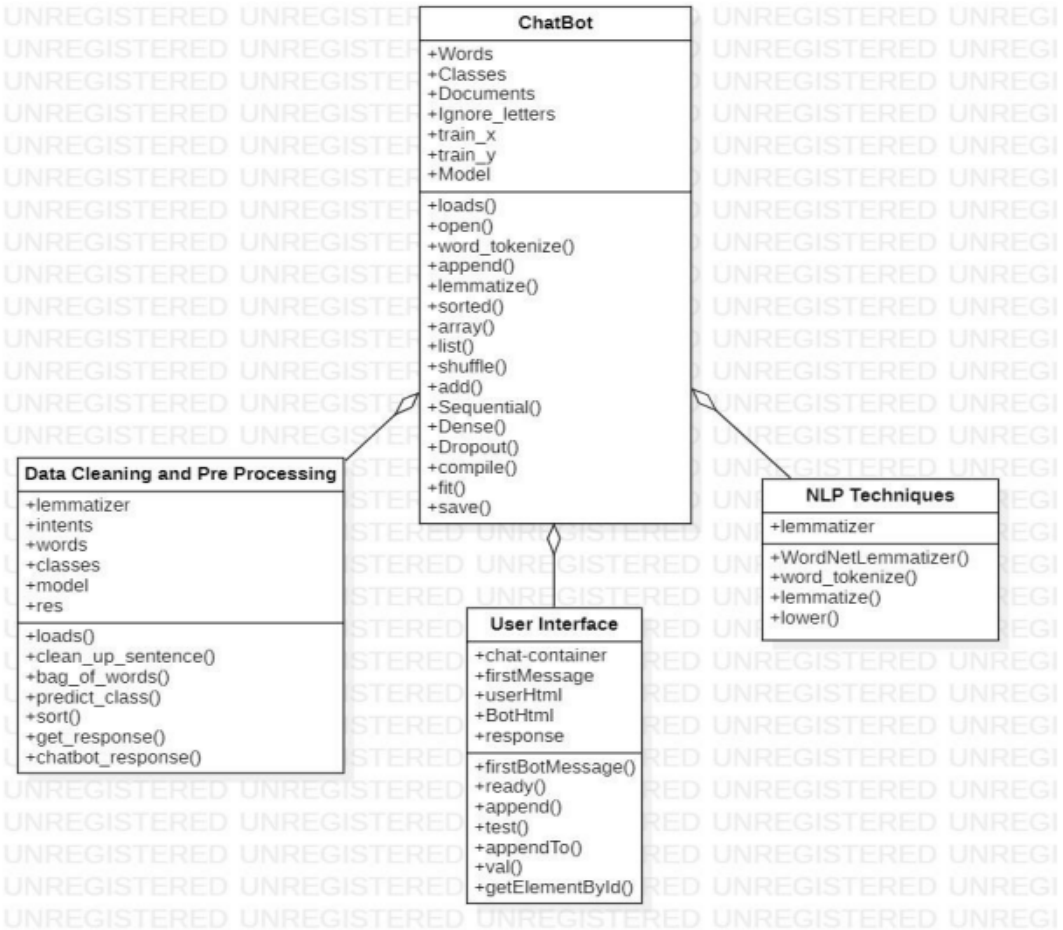
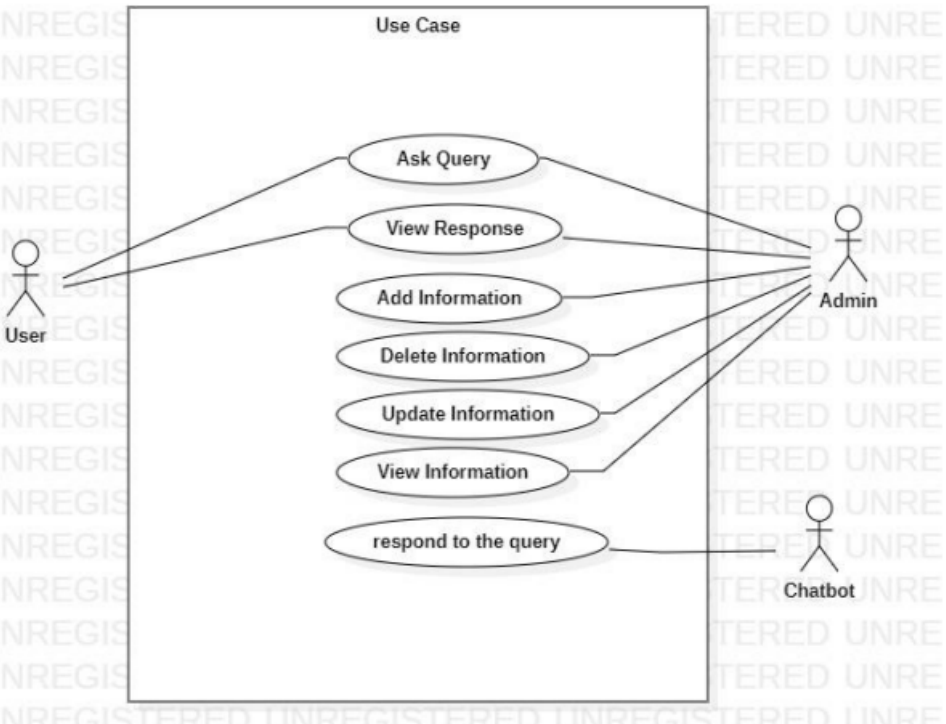


Figure 11 : Class Diagram of Chatbot

Four classes—ChatBot, NLP Techniques, Data Cleaning and Preprocessing, and User Interface—are depicted in the class diagram in Figure 11. Preprocessing, training the model, and loading data from the intents file are the tasks assigned to the ChatBot class. Preprocessing input text and response prediction are handled by data cleaning and preprocessing. Text lemmatization is handled by the NLP Techniques class, and input data extraction and user presentation are handled by the User Interface class.

6.6. USECASE DIAGRAM

A use case diagram shows how a system behaves dynamically. It incorporates actors, use cases, and their subsystem inside an application need are modeled in this diagram. It displays a system's high-level functionality and shows how users engage with it.



.

Figure 12: Use Case Diagram

The user, administrator, and chatbot are the three main actors in the suggested system, as the diagram shows. It is the user's responsibility to pose queries and read the answers. In contrast, the administrator has the ability to read responses, ask questions, and administer the database by adding, removing, and viewing data. Lastly, the chatbots job is to process the questions and give the best possible responses.

6.7. ACTIVITY DIAGRAM

Activity diagrams describe the procedures followed to carry out a use case and demonstrate how control moves through a system. They assist us in simulating concurrent and sequential tasks. Activity diagrams essentially give workflows a visual representation. By charting the control from a beginning point to an endpoint and emphasizing the various decision routes that may emerge during the process, it demonstrates what causes a particular event.

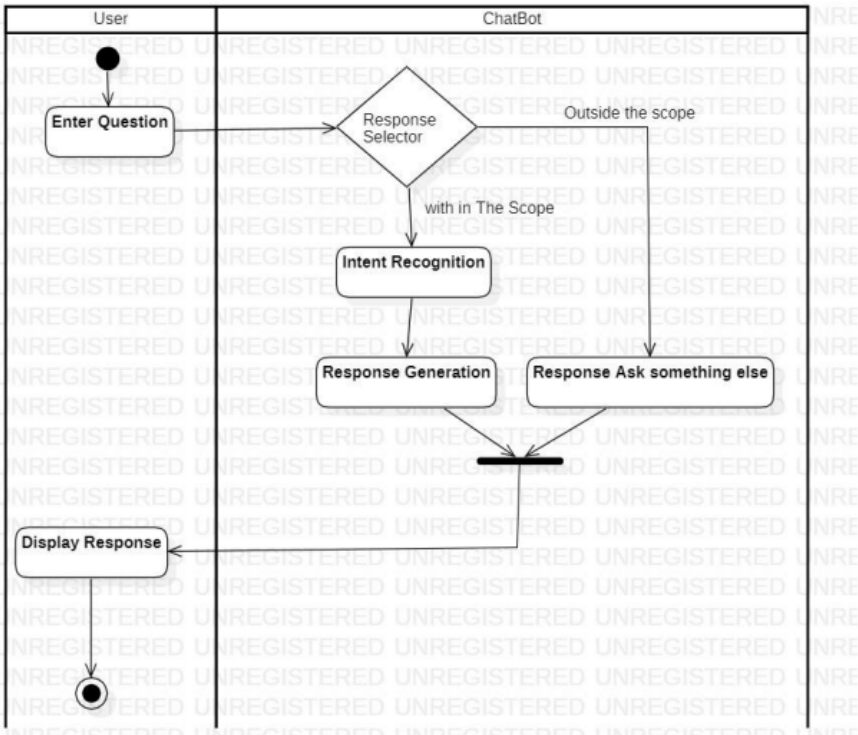


Figure 13: Activity Diagram

The system's activity diagram is depicted in the diagram above. The user and the chatbot are the two companies engaged. While the chatbot is in charge of creating answers to user inquiries, the user is in charge of asking questions and viewing answers.

6.8. SEQUENCE DIAGRAM:

The communication between two lifelines is depicted in this figure as a time-ordered series of that take place throughout runtime. In UML, a vertical bar represents a lifeline, and a vertical dotted line that extends across the bottom of the page represents the message flow. It incorporates branching as well as iterations. In essence, a sequence diagram shows how things interact with one another in a particular order and at what times.

Sequence diagrams can also be referred to as event diagrams or event scenarios. These diagrams illustrate the order in which various components of a system function. Software engineers.

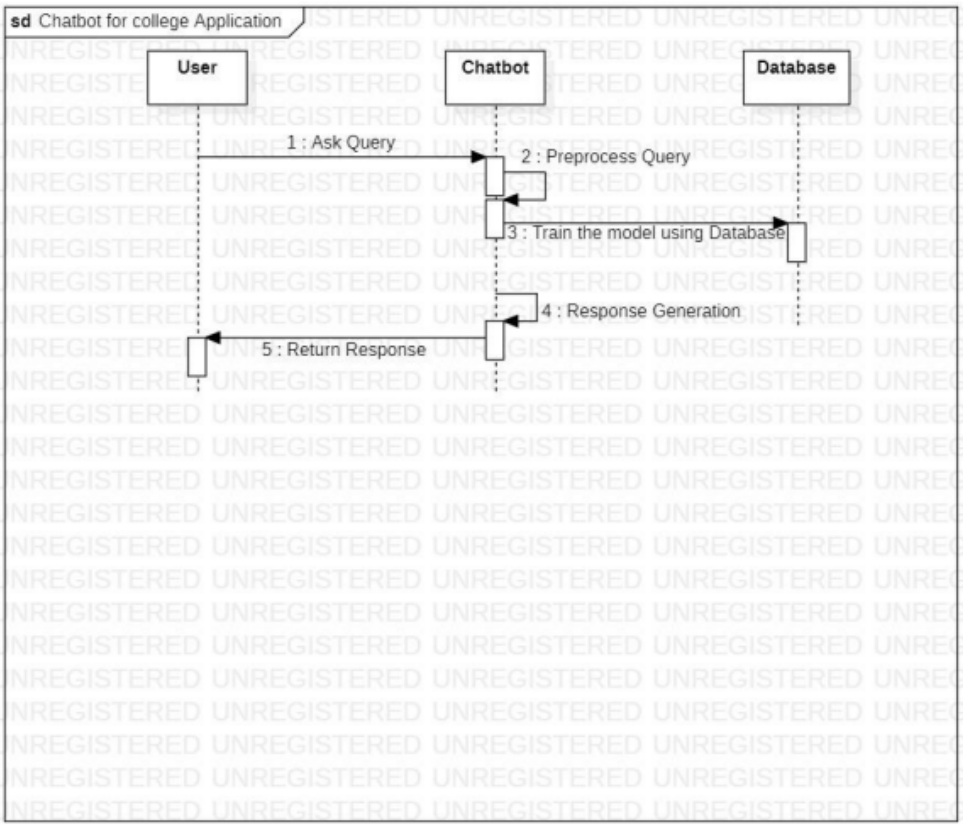


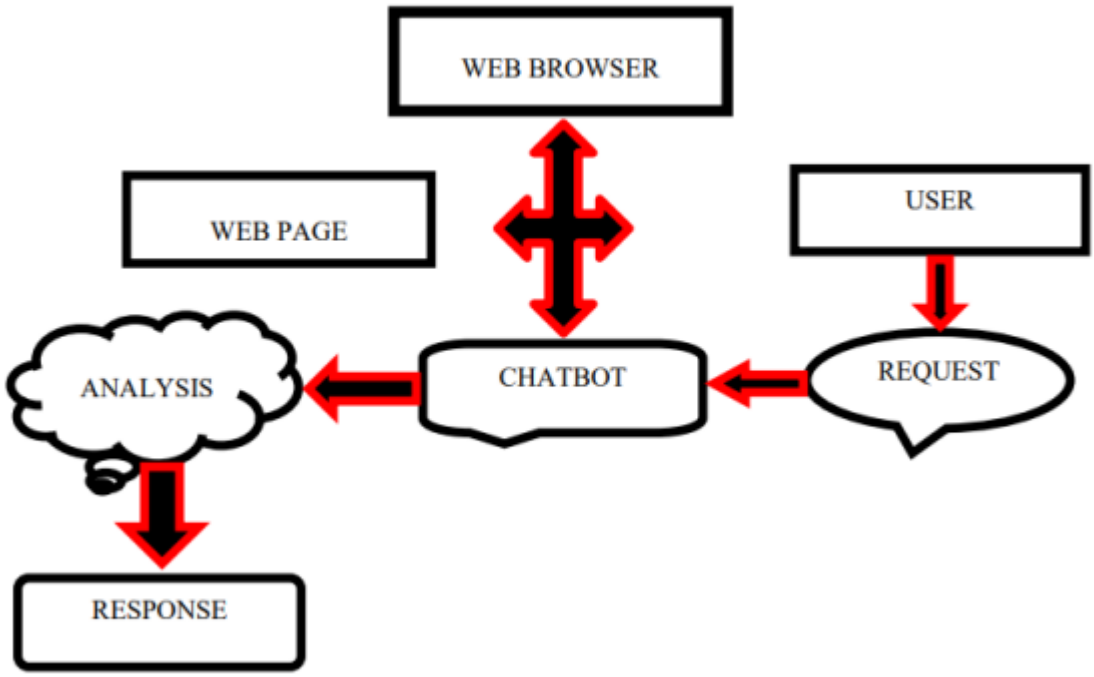
Figure 14 : Sequence Diagram

Figure 14. Three LifeLines are depicted in this sequence diagram: database, chatbot, and user. The chatbot receives a query that the user provides. The chatbot uses the database to train the model after preprocessing the data. The chatbot then looks up the relevant answer in the database and preprocesses the question. Lastly, the user receives the response from the chatbot.

6.9. : SYTEM ARCHITECTURE

The general architecture of the suggested system is described in this section. This chatbot's main objective is to respond to user questions without requiring human assistance. The chatbot is accessible to users via any web browser. The chatbot receives a request from a user, processes it, and responds. A variety of machine learning algorithms

are used in this processing. Specific tags, which are basically keywords that help the chatbot comprehend the user's request, are linked to each query. The chatbot provides the relevant response after evaluating the input. When a user makes an unclear request, the chatbot automatically responds with pre-programmed standard replies. With very few exceptions, the majority of user inquiries are answered in a straightforward manner.



.

Figure 15 : System Architecture

6.9.1 IMPLEMENTATION

This provides a broad overview of the system (on an overall basis) and a comprehensive version of the chatterbot's software and configured query dataset. Included is the algorithmic process, which is followed by the system design motivation. JavaScript, HTML, CSS, and Python are used in this implementation. These library features will assist the chatbot in comprehending the user's request and the appropriate answer. Additionally, chatbot has its own Python module, which is mostly necessary for developing a user-friendly chatbot. [1].

Figure 1 shows that Chatbot Development flow chart. In this flow chart User input the query and it will pass through different steps to generate output. From beginnings when the user giving a query as input like Text. After that the text will moved forwarded to Convert text as Lowercase letter. Next step is tokenization process in which all the text should be divided in sub-words and send to next step. In this step the program should check if the sentence or word are meaningful then if generate output for the next step. The next step is stemming it is the process of checking the prefix, suffix or infix. This check the grammatical uses in sentence is correct or not. If the grammatical is not in proper form, then it shows error. After all step’s passed the program produce output and again ask Any New Input or End the task. It will help users to ask the query and solve the problem. It is very fast to find the solution, if the solution is not in the database it automatically

updates the data. It shows the data in easiest way like tabular form, numbering, using pointers.

function and applied stochastic gradient descent (SGD) to arrive at the correct weights. Additionally, accuracy is the statistic we have selected. To get the needed accuracy, we will train the Python chatbot model roughly 200 times.These are the functions:

1. The first core component is Natural Language understanding in this step it checks the Topic Detection, Intent Analysis and Entity Linking. First we are talking about Topic Detection in this topic can be selected and check if the topic should already present or not. If it is already present, then it shows error otherwise its correct. In the second step Intent analysis it checks every word is connected to other words/sentence or not.
2. When a user submits a sentence, it checks the sentence and check the content is already present or not in which if the content is already present it shows error otherwise it passes to the next step.
3. The Dialogue Manager (DM) is another essential component of any chatbot, it is of two types Rule based and Knowledge based. In Figure 3 shows that if Rule Based is Yes then everything is correct and its ready to move for next step, if No it is move to knowledge based. Then again checks in the knowledge base process if yes it will move to

next step otherwise it moved to Retrieval Based and Generative Based.

4. The last major core component of any chatbot. It is of two types Content Filter and Engagement Ranking. In content filter Natural Language Generator checks the content of the project is the content is already taken or present it reject the project. Give a pop up message to its already present choose new one. And the second step is Engagement Ranking if the article/report is passed to next step it uploaded to database and it checks the similarity and Ranking.

6.9.2 ALGORIHTM

AI chatbots can be developed using a variety of algorithms. The method based on natural language processing is the most popular of all of those. When it comes to the chatbot's quality, text processing, classification, and interpretation become crucial since it receives input in natural language.

Popular chatbot algorithms include the following ones:

1. Naive Bayes Algorithm

2. Support vector Machine

3. Natural language processing

4. Recurrent neural networks

5. Long short-term memory

6. Markov models

1. Naive Bayes algorithm:

In order to reduce the number of possible answers, the Naive Bayes algorithm attempts to classify text in order to deduce the cause of the action. Since determining the goal of a discussion between a chatbot and a person is one of the first steps in the process, this algorithm is quite important. Certain words have a value in some categories, and their value might vary with frequency within the categories because the algorithm depends on commonality. It makes it possible to categorize text data based on both phrase and intent.

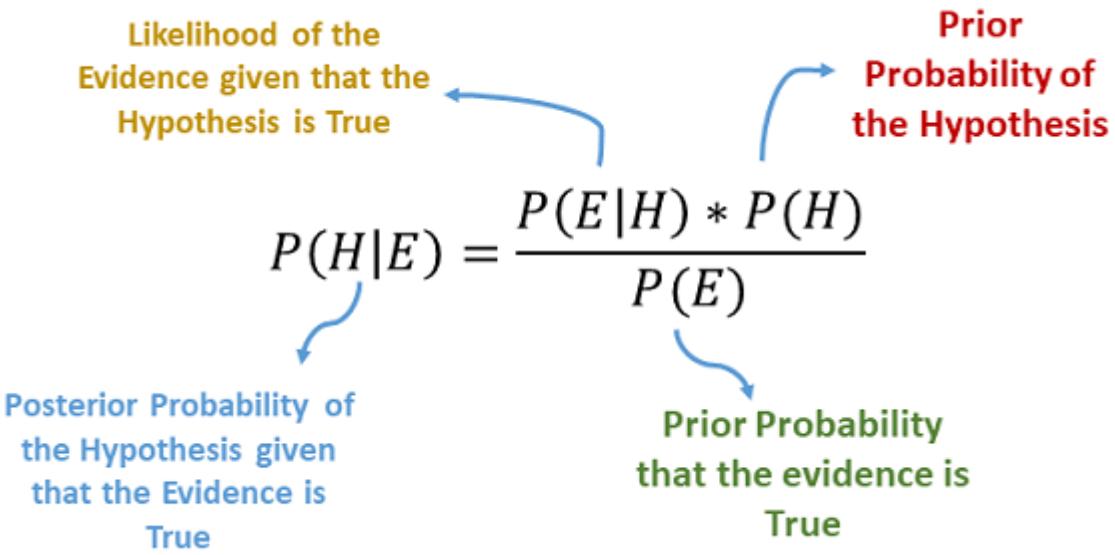


Figure 16: Naive Bayes algorithm

2. Support vector machine:

Support Vector Machines (SVMs) are grounded in the Structural Risk Minimization Principle. They perform exceptionally well with text data and chatbots due to the high-dimensional input space created by numerous text features, the linear separability of data, and the frequent occurrence of sparse matrices. This makes SVMs one of the most popular algorithms for text classification and sentiment analysis.

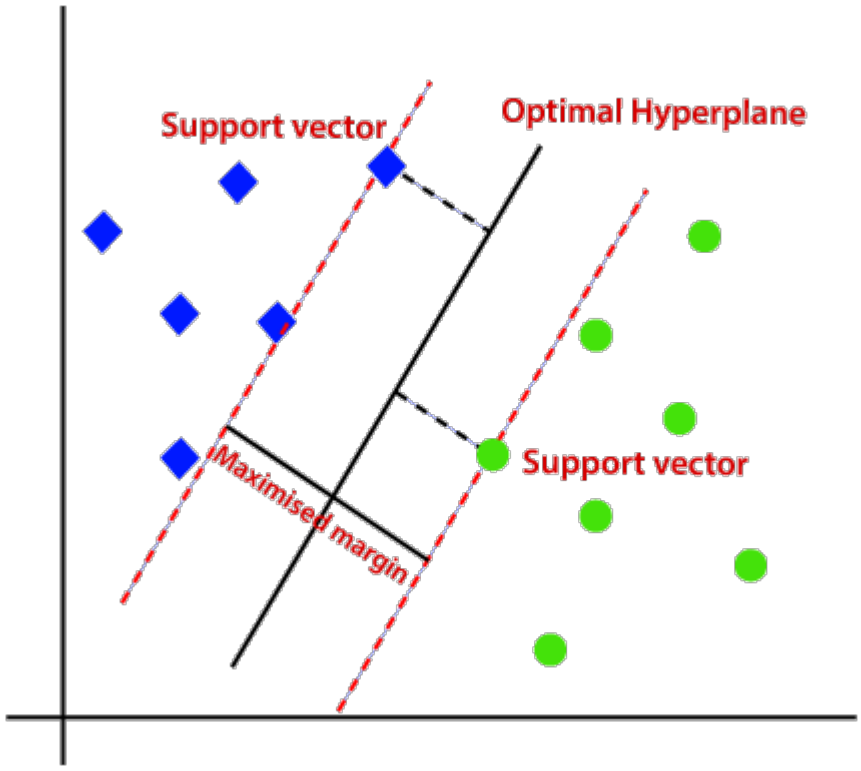


Figure 17: Support Vector Machine

3. Natural Language processing algorithms:

Because it specifies how the bot will comprehend and interpret the text input, natural language processing (NLP) is very important for chatbots. The consumer would not even be aware that they were interacting with a machine in the ideal chatbot's interactions. Using a wealth of conversational data and machine learning, this program aims to understand the nuances of human language. Because it can read grammar, sentiment, and intent in text data, the bot gains from natural language processing.

4) Recurrent Neural Networks:

Neural networks that can process sequential data to capture the context of words in a particular text input are known as recurrent neural networks. By iterating through the sequence elements and keeping a state containing information pertaining to what it has already observed, RNN processes the text input in a manner similar to that of biological intelligence. Consequently, we are able to understand and document the input's context.

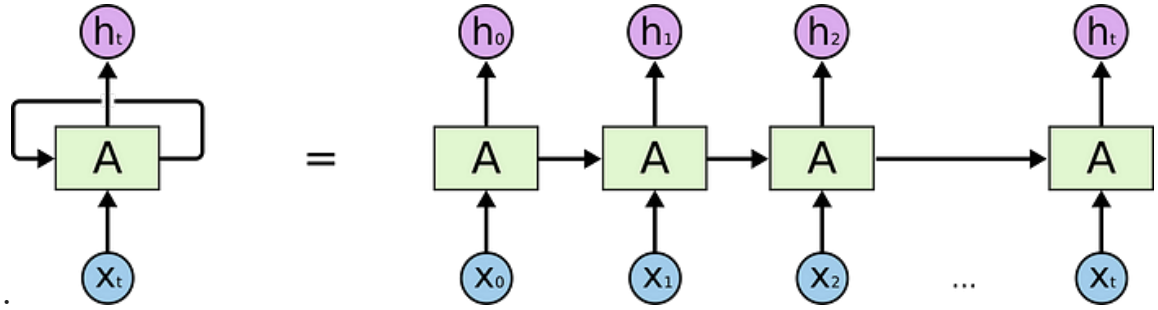


Figure 18: Recurrent Neural Networks

5) Long short-term memory:

LSTMs, while similar to recurrent neural networks (RNNs), outperform them because they are designed to retain and learn from the states of earlier inputs. Unlike RNNs, they maintain memories of

previous inputs in a sequence, making them well-suited for managing lengthy sequences. This feature makes LSTMs particularly useful in conversational AI for tasks like predicting the next word.

6) Markov models for text generation

Markov chains are used in text production and chatbots. They operate by calculating the likelihood of changing states. This model is simple to use and summarize because it can be conveniently stored as matrices. Instead of taking into account the path to get there, these chains use the previous state to determine the current state.

7) Grammar and Parsing Algorithms

Grammar and parsing algorithms are able to recognize and eliminate ambiguity in sentences. In order to guarantee that the chatbot will converse without grammatical errors, the grammar algorithm provides a formal specification of a language's structure. The parsing algorithm checks the sentence's grammatical structure using the grammar.

6.5.5. SYSTEM STUDY AND TESTING

A system study is a thorough examination that is carried out very early in a project to ascertain how the current system operates, where its limitations are, and what the needs of its users are. To learn about the system's goals and potential issues, interviews, observations, and

examinations of documents from all sources are gathered. The main goal is to precisely characterize the current system that may be utilized to improve the solution in line with the organization's needs and user expectations. Within a project, the system research includes a number of crucial tasks:

- **Requirement Gathering**: Collecting detailed information on what users need from the new system.
- **System Analysis**: Examining the current system to identify its strengths, weaknesses, and areas for improvement.
- **Feasibility Study**: Assessing the practicality and viability of proposed solutions.
- **Documentation**: Recording findings and requirements to guide the subsequent design and development phases.

Fully integrated systems undergo system testing at this crucial step of the software development lifecycle to make sure that the criteria are being met. In order to ensure proper functionality and performance, it evaluates the entire software application against both functional and nonfunctional standards.

It is carried out before to acceptance testing and following integration testing. One kind of black-box testing looks at the system's exterior behavior instead of its internal code structure.
The following are the main goals of system testing:

- **Validating End-to-End Scenarios**: Ensuring that the system performs as intended across various use cases.
- **Identifying Defects**: Detecting issues that may not have been uncovered during earlier testing phases.

- **Assessing Compliance**: Verifying that the system adheres to business and technical requirements.
- **Evaluating Performance**: Measuring system responsiveness, stability, and resource utilization under different conditions.

CHAPTER-7

TIMELINE FOR EXECUTION OF PROJECT

(GANTT CHART)

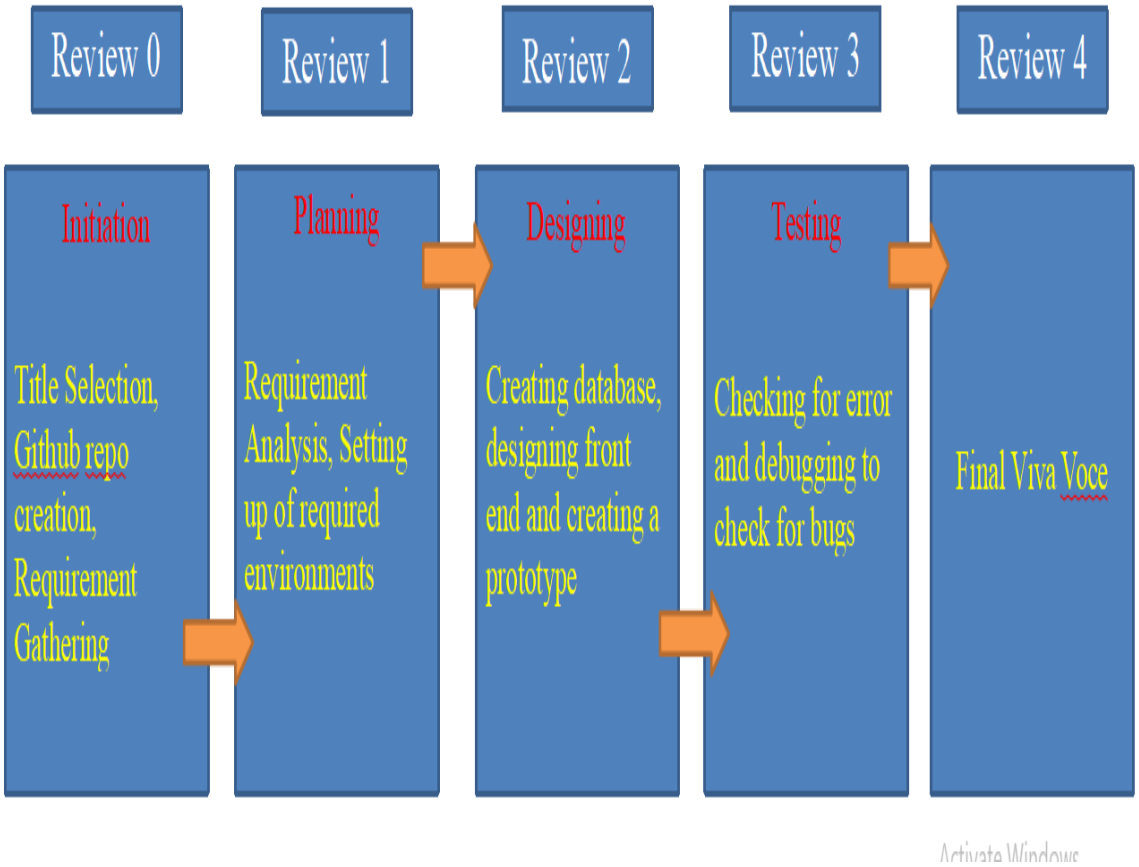


Figure 20: Timeline

CHAPTER-8

OUTCOMES

1. Customer Support

- **Positive Outcomes:**
  - Faster resolution of customer queries.
  - 24/7 availability for answering frequently asked questions.
  - Reduced workload for human agents, allowing them to handle complex queries.
  - Improved customer satisfaction and engagement.
- **Negative Outcomes:**
  - Frustration if the chatbot fails to understand the query.
  - Limited capabilities for handling nuanced or emotional conversations.

2. Sales and Marketing

- **Positive Outcomes:**
  - Increased lead generation through proactive engagement.
  - Enhanced user experience with personalized recommendations.
  - Automation of product or service inquiries.
- **Negative Outcomes:**
  - Poor engagement if the chatbot lacks personalization or fails to build trust.
  - Loss of potential leads if the bot's responses are irrelevant or unhelpful.

3. Healthcare

- **Positive Outcomes:**
  - Quick access to health-related information and resources.
  - Support for appointment scheduling and reminders.
  - Assistance in triaging symptoms and guiding users to appropriate care.
- **Negative Outcomes:**
  - Risk of providing inaccurate or unsafe advice.
  - User distrust if the chatbot lacks transparency about its limitations.

4. Education

- **Positive Outcomes:**
  - Improved learning experience with interactive and personalized tutoring.
  - Access to educational content at any time.
  - Support for language learning and skill development.
- **Negative Outcomes:**
  - Reduced engagement if the bot isn't interactive or responsive enough.
  - Difficulty addressing diverse learning needs without advanced AI.

5. E-commerce

- **Positive Outcomes:**
  - Simplified shopping experience with product recommendations.
  - Efficient handling of order tracking and return processes.
  - Enhanced customer loyalty through personalized interactions.
- **Negative Outcomes:**
  - Frustration from incorrect or irrelevant suggestions.
  - User dissatisfaction if the bot cannot escalate issues effectively.

CHAPTER-9
RESULTS AND DISCUSSIONS

Feedback from testing the final prototype indicated that participants were not as confused by information from a chatbot as they were by information from a human. The fact that the chatbot cited a source for the information it presented may have contributed to the rating of the material as not being less reliable. Examining the participants' reactions to the chatbot and their subsequent reports on it has been fascinating. The results show some signs that a chatbot can be a good substitute for a supportive companion that new students might require at their new school.

However, it is important to note that the evaluators had to adapt their language to the chatbot's comprehension because the chatbots were not particularly sophisticated.

We were unable to conduct as much user testing and redesign the chatbot as we would have liked due to the project's scale. This has an effect on the validity of our research. Since we discussed this when creating the chatbot, it should also have a greater focus throughout the project to increase its legitimacy. We cannot say that consumers trust a chatbot as much as they trust a human being, even if the participants trusted the information in this project. One of the biases in our study is that all of the students who were involved in it were already familiar with many of the answers that the prototype could offer.

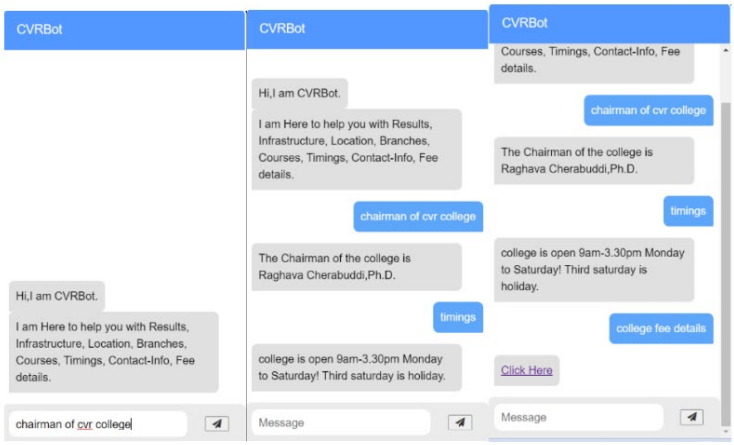


Figure 21 : Text Based Responses

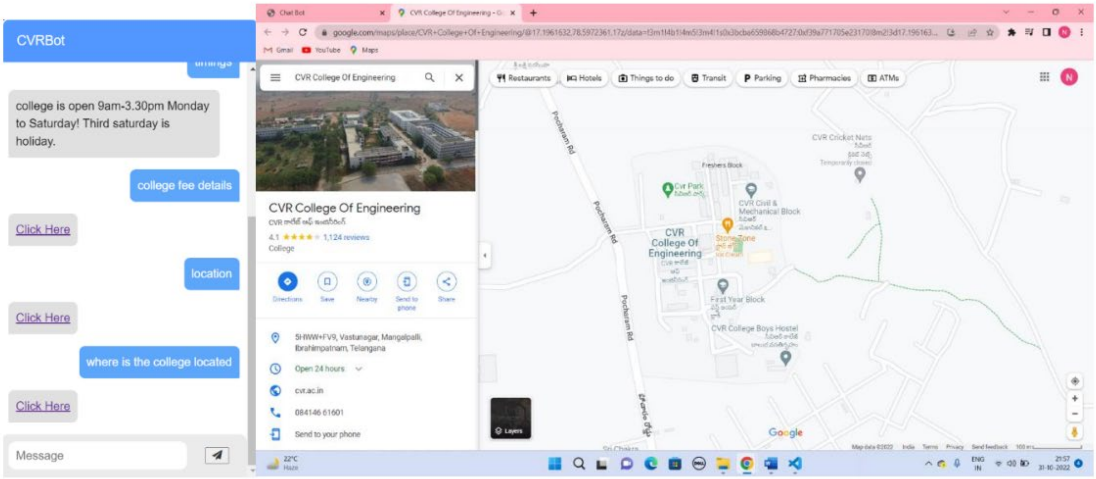


Figure 22 : Response in the form of URL

CHAPTER-10

CONCLUSION

The project's objective is to decrease the number of employees while increasing the speed at which user inquiries are answered. In the past, users would submit an inquiry email to the specific site administrator, and it would take a few days for the individual to respond. This delay can be avoided by using chatbots, which provide prompt, pertinent answers to user requests or queries. In order to respond to customer requests more quickly, chatbots have been built for the majority of websites in the banking, educational, and business sectors. Chatbots are artificially intelligent devices that are easy to operate. By including speech recognition and multiple languages, this idea can be further enhanced. As the website is built, we can add a lot more tags to the data collection. After a chat session ends, this program can send an email with the user's chat history. Making users approved and obtaining their email addresses will do this. With a farmer-centric approach, multilingual support, real-time notifications, and user-friendly navigation, a unified platform integrates all these features into a single mobile and web application. Based on user testing, the system was shown to be scalable, easily accessible, and beneficial for smallholder farmers in remote regions. Because backend design is powered by cloud computing, it guarantees seamless API interaction, effective data processing, and scalability to accommodate a variety of crops and diseases in different geographical areas.

It only takes a little work to make the website easy to use and understand for users.

REFERENCES

[1] D. P. Shanmugapriya, Harini M., "Development of Chatterbot using Python", International Journal of Computer Applications (0975 – 8887) Volume 176 – No. 21, May 2020,pp 18-20.

[2] Harshala Gawade, Prachi Vishe, Vedika Patil, Sonali Kolpe, "College Enquiry ChatBot System", International Journal of Engineering Research & Technology (IJERT), ISSN: 2278-0181,Vol. 9 Issue 09, September-2020, pp 741-742.

[3] Ms.Ch. Lavanya Susanna, R. Pratyusha, P. Swathi, P. Rishi Krishna, V. Sai Pradeep, "COLLEGE ENQUIRY CHATBOT" , International Research Journal of Engineering and Technology (IRJET), e-ISSN: 2395-0056, p-ISSN: 2395-0072, Volume: 07 Issue: 3 | Mar 2020,pp 784-788.

[4] Hrushikesh Koundinya K, Ajay Krishna Palakurthi, Vaishnavi Putnala, "SMART COLLEGE CHATBOT USING ML AND PYTHON", 10.1109/ICSCAN49426.2020. 9262426, December 2020.[Online] Available "https://www.researchgate.net/publication/347423315".

[5] Gandhar Khandagale, Meghana Wagh, Pranali Patil, Prof. Satish Kuchiwale ,"INTELLIGENT CHATBOT FOR COLLEGE ENQUIRY SYSTEM",International Research Journal of Engineering and Technology, e-ISSN: 2395-0056, p-ISSN: 2395- 0072, Volume: 09 Issue: 04 | Apr 2022, pp 2400-2404.

[6] Emanuela Haller, Traian Rebedea, “Designing a Chat-bot that Simulates an Historical Figure” , 2013 19th International Conference on Control Systems and Computer Science,2013, [Online]. Available https://ieeexplore.ieee.org/document/6569323.

[7] Maria Jo˜ao Pereira, Lu´ısa Coheur, “Just.Chat - a platform for processing information to be used in chatbots”,May 2013, [Online] ,Available “https://www.inescid.pt/publications/7307/pdf”.

[8] M. Rahman, A. A. Mamun and A. Islam, "Programming challenges of chatbot: Current and future prospective," 2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC), Dhaka, 2017,pp.75-78. doi: 10.1109/R10-HTC.2017.8288910.

[9] B. Setiaji and F. W. Wibowo, "Chatbot Using a Knowledge in Database: Human-toMachine Conversation Modeling," 2016, 7th International Conference on Intelligent Systems, Modeling and Simulation (ISMS), Bangkok, pp. 72-77.

[10] https://python.plainenglish.io/create-a-deep-learning-chatbot-with-python-and-flaskd75396a4382a

APPENDIX-A

PSEUDOCODE

```
from sklearn.tree import DecisionTreeClassifier
DecisionTree =
DecisionTreeClassifier(criterion="entropy",random_state=2,max_depth=5)
DecisionTree.fit(Xtrain,Ytrain)
predicted_values = DecisionTree.predict(Xtest)
x = metrics.accuracy_score(Ytest, predicted_values)
acc.append(x)
model.append('Decision Tree')
print("DecisionTrees's Accuracy is: ", x*100)
print(classification_report(Ytest,predicted_values))

from sklearn.ensemble import RandomForestClassifier
RF = RandomForestClassifier(n_estimators=20, random_state=5)
RF.fit(Xtrain,Ytrain
predicted_values = RF.predict(Xtest)
x = metrics.accuracy_score(Ytest, predicted_values)
acc.append(x)
model.append('RF')
print("RF's Accuracy is: ", x)
print(classification_report(Ytest,predicted_values))

from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score, classification_report
classifier = KNeighborsClassifier(n_neighbors=5, metric='minkowski', p=2)
classifier.fit(Xtrain, Ytrain)
```

```
y_pred = classifier.predict(Xtest)
accuracy = accuracy_score(Ytest, y_pred)
acc.append(accuracy)
model.append('KNN')
print("KNN classifier's Accuracy is:", accuracy)
print(classification_report(Ytest, y_pred))

import numpy as np
import tensorflow as tf
from tensorflow import keras
from keras._tf_keras.keras.preprocessing.image import ImageDataGenerator
import matplotlib.pyplot as plt
validation_set = keras.utils.image_dataset_from_directory('Dataset1/valid',
    labels="inferred",
    label_mode="categorical",
    class_names=None,
    color_mode="rgb",
    batch_size=32,
    image_size=(128, 128),
    shuffle=True,
    seed=None,
    validation_split=None,
    subset=None,
    interpolation="bilinear",
    follow_links=False,
    crop_to_aspect_ratio=False
)
class_name = validation_set.class_names
print(class_name)
```

APPENDIX-B

SCREENSHOTS

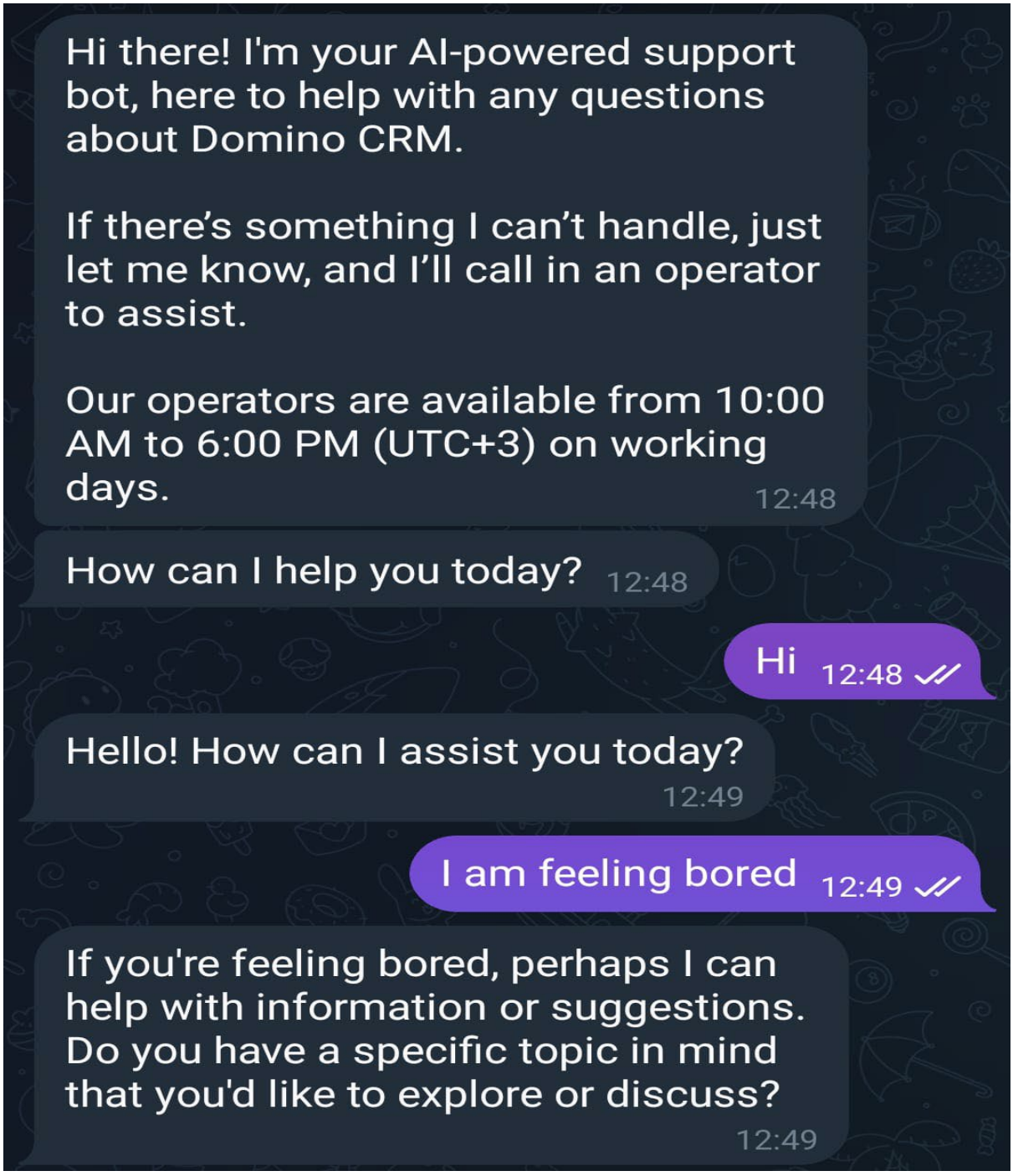


Figure 23. It shows that Chatbot used in mobile

```
chatbot()
```

```
Me: hey!
Chabot: hey what's up
Me: nothing how are you
Chabot: pretty good, just doing a lot of work
Me: i hate you
Chabot: hahaha aww nah you like me
Me:
```

Figure 24: User Interactions

```
Me: hey
Chatbot: hi


Me: lets go out
Chatbot: i wanted to like 20 minutes ago but now i'm like dead gaga


Me: please
Chatbot: yeah no


Me: where is the center of the maze dolores?
Chatbot: on allen st right in les


Me:
```

Figure 25: How Chatbot solving customer problem

APPENDIX-C
ENCLOSURES

The Project work carried out here is mapped to SDG-3 Good Health and Well-Being.

The project work carried here contributes to the well-being of the human society. This can be used for Analyzing and detecting blood cancer in the early stages so that the required medication can be started early to avoid further consequences which might result in mortality.

PUBLICATIONS

https://ijsrem.com/download/development-and-implementation-of-chatbot-using-machine-learning/

ISSN: 2582-3930
Impact Factor: 8.448

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING & MANAGEMENT

An Open Access Scholarly Journal || Index in major Databases & Metadata

CERTIFICATE OF PUBLICATION

International Journal of Scientific Research in Engineering & Management is hereby awarding this certificate to

Dr. Iqbal Gani Dar

in recognization to the publication of paper titled

Development and Implementation of Chatbot Using Machine Learning

published in IJSREM Journal onVolume 09 Issue 01 January, 2025

www.ijsrem.com

Editor-in-Chief
IJSREM Journal

e-mail: editor@ijsrem.com

ISSN: 2582-3930
Impact Factor: 8.448

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING & MANAGEMENT
An Open Access Scholarly Journal || Index in major Databases & Metadata

CERTIFICATE OF PUBLICATION

International Journal of Scientific Research in Engineering & Management is hereby awarding this certificate to

Chakshu

in recognization to the publication of paper titled

Development and Implementation of Chatbot Using Machine Learning

published in IJSREM Journal onVolume 09 Issue 01 January, 2025

Editor-in-Chief
IJSREM Journal

www.ijsrem.com

e-mail: editor@ijsrem.com

ISSN: 2582-3930
Impact Factor: 8.448

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING & MANAGEMENT

An Open Access Scholarly Journal || Index in major Databases & Metadata

CERTIFICATE OF PUBLICATION

International Journal of Scientific Research in Engineering & Management is hereby awarding this certificate to

Choleshwar Kumar

in recognization to the publication of paper titled

Development and Implementation of Chatbot Using Machine Learning

published in IJSREM Journal onVolume 09 Issue 01 January, 2025

Editor-in-Chief
IJSREM Journal

www.ijsrem.com

e-mail: editor@ijsrem.com

ISSN: 2582-3930
Impact Factor: 8.448

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING & MANAGEMENT

An Open Access Scholarly Journal || Index in major Databases & Metadata

CERTIFICATE OF PUBLICATION

International Journal of Scientific Research in Engineering & Management is hereby awarding this certificate to

Amarjeet Kumar

in recognization to the publication of paper titled

Development and Implementation of Chatbot Using Machine Learning

published in IJSREM Journal onVolume 09 Issue 01 January, 2025

www.ijsrem.com

Editor-in-Chief
IJSREM Journal

e-mail: editor@ijsrem.com

ISSN: 2582-3930
Impact Factor: 8.448

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING & MANAGEMENT

An Open Access Scholarly Journal || Index in major Databases & Metadata

CERTIFICATE OF PUBLICATION

International Journal of Scientific Research in Engineering & Management is hereby awarding this certificate to

Komma Bhanu Prakash Reddy

in recognization to the publication of paper titled

Development and Implementation of Chatbot Using Machine Learning

published in IJSREM Journal onVolume 09 Issue 01 January, 2025

www.ijsrem.com

Editor-in-Chief
IJSREM Journal

e-mail: editor@ijsrem.com

ISSN: 2582-3930
Impact Factor: 8.448

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING & MANAGEMENT

An Open Access Scholarly Journal || Index in major Databases & Metadata

CERTIFICATE OF PUBLICATION

International Journal of Scientific Research in Engineering & Management is hereby awarding this certificate to

Samrat Sarkar

in recognization to the publication of paper titled

Development and Implementation of Chatbot Using Machine Learning

published in IJSREM Journal onVolume 09 Issue 01 January, 2025

www.ijsrem.com

Editor-in-Chief
IJSREM Journal

e-mail: editor@ijsrem.com

ORIGINALITY REPORT

12%	8%	6%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.cis.upenn.edu Internet Source	2%
2	www.jetir.org Internet Source	1%
3	medium.com Internet Source	1%
4	Dilyan Grigorov. "Chapter 1 Evolution and Significance of Large Language Models", Springer Science and Business Media LLC, 2024 Publication	1%
5	Vijayaraghavan V., Jack Brian Cooper, Rian Leevinson J.. "Algorithm Inspection for Chatbot Performance Evaluation", Procedia Computer Science, 2020 Publication	1%
6	Submitted to Arab Open University Student Paper	1%
7	Submitted to Presidency University Student Paper	<1%

8	Submitted to Pathfinder Enterprises Student Paper	<1 %
9	Submitted to University of Greenwich Student Paper	<1 %
10	ijsrst.com Internet Source	<1 %
11	Submitted to Federation University Student Paper	<1 %
12	Natasa Kleanthous, Abir Hussain. "Machine Learning in Farm Animal Behavior using Python", CRC Press, 2025 Publication	<1 %
13	Submitted to Southampton Solent University Student Paper	<1 %
14	Submitted to City University Student Paper	<1 %
15	H.L. Gururaj, Francesco Flammini, S. Srividhya, M.L. Chayadevi, Sheba Selvam. "Computer Science Engineering", CRC Press, 2024 Publication	<1 %
16	aiforsocialgood.ca Internet Source	<1 %
17	Submitted to King's Own Institute Student Paper	<1 %

18	www.analyticsvidhya.com Internet Source	<1 %
19	Submitted to Singapore Institute of Technology Student Paper	<1 %
20	Suneeta Satpathy, Bijay Kumar Paikaray, Ming Yang, Arun Balakrishnan. "Sustainable Farming through Machine Learning - Enhancing Productivity and Efficiency", CRC Press, 2024 Publication	<1 %
21	www.ijirset.com Internet Source	<1 %
22	E.J. Klein. "Automatic classification of chromatographic peaks", Proceedings of the 1997 American Control Conference (Cat No 97CH36041) ACC-97, 1997 Publication	<1 %
23	www.technologistsinsync.com Internet Source	<1 %
24	isteonline.in Internet Source	<1 %
25	makingaieasy.com Internet Source	<1 %
26	www2.mdpi.com Internet Source	<1 %

27	Mark Chang. "Artificial Intelligence for Drug Development, Precision Medicine, and Healthcare", CRC Press, 2020 Publication	<1 %
28	Pawan Singh Mehra, Dhirendra Kumar Shukla. "Artificial Intelligence, Blockchain, Computing and Security - Volume 2", CRC Press, 2023 Publication	<1 %
29	Sameer Qazi, Muhammad Bilal Kadri, Muhammad Naveed, Bilal A. Khawaja et al. "AI-Driven Learning Management Systems: Modern Developments, Challenges and Future Trends during the Age of ChatGPT", Computers, Materials & Continua, 2024 Publication	<1 %
30	ijrpr.com Internet Source	<1 %
31	www.gpcet.ac.in Internet Source	<1 %
32	www.kinokuniya.co.jp Internet Source	<1 %
33	Anurag Tiwari, Manuj Darbari. "Emerging Trends in Computer Science and Its Application - Proceedings of the International Conference on Advances in Emerging Trends in Computer Applications (ICAETC-2023)	<1 %

December 21–22, 2023, Lucknow, India", CRC Press, 2025
Publication

34 Soyeon Caren Han, Henry Weld, Yan Li, Jean Lee, Josiah Poon. "Natural Language Understanding in Conversational AI with Deep Learning", Springer Science and Business Media LLC, 2025 <1 %
Publication

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		