SELF LEARNING BOT

A PROJECT REPORT

Submitted by,

MARTALA MAHESWAR REDDY - 20201CAI0104
SALAPAKSHI GANESH - 20201CAI0096
KARTHIK SHARMA - 20201CAI0131
PAMU ARUN TEJA - 20201CAI0152

Under the guidance of,
Dr. ASIF MOHAMMED H.B

in partial fulfillment for the award of the degree of
BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

At

PRESIDENCY UNIVERSITY
BENGALURU
JANUARY 2024

SELF LEARNING BOT

A PROJECT REPORT

Submitted by,

MARTALA MAHESWAR REDDY - 20201CAI0104
SALAPAKSHI GANESH - 20201CAI0096
KARTHIK SHARMA - 20201CAI0131
PAMU ARUN TEJA - 20201CAI0152

Under the guidance of,
Dr. ASIF MOHAMMED H.B

in partial fulfillment for the award of the degree of
BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

At

PRESIDENCY UNIVERSITY
BENGALURU
JANUARY 2024

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE ENGINEERING & INFORMATION SCIENCE

CERTIFICATE

This is to certify that the Project report **"SELF LEARNING BOT"** being submitted by "MARTALA MAHESWAR REDDY , SALAPAKSHI GANESH , ALURU KARTHIK SHARMA , PAMU ARUN TEJA" bearing roll numbers "20201CAI0104 , 20201CAI0096 , 20201CAI0131 , 20201CAI0152 " in partial fulfilment of requirement for the award of degree of Bachelor of Technology in **Computer Science and Engineering** is a bonafide work carried out under my supervision.

Dr. ASIF MOHAMMED H.B
Assistant Professor
School of CSE&IS
Presidency University

Dr . ZAFAR ALI KHAN N
Associate Professor & HOD
School of CSE&IS
Presidency University

Dr . C. KALAIARASAN
Associate Dean
School of CSE&IS
Presidency University

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

Dr . SAMEERUDDIN KHAN
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 **SELF LEARNING BOT** in partial fulfilment 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 **Prof . Dr . ASIF MOHAMMED H.B , School of Computer Science Engineering , Presidency University, Bengaluru.**

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

NAMES	ROLLNO	SIGNATURE
MAHESWAR REDDY	20201CAI0104	
S GANESH	20201CAI0096	
KARTHIK SHARMA	20201CAI0131	
ARUN TEJA	20201CAI0152	

ABSTRACT

Natural Language Processing (NLP) has undergone a transformative evolution in recent years, revolutionizing the way humans interact with computers. This project introduces a Chatbot Application that is intricately designed to leverage cutting-edge NLP techniques within a user-friendly Tkinter - based graphical user interface. The primary objective of this project is to create an intelligent, responsive, and versatile assistant capable of handling diverse user inputs through both text and voice.

The Chatbot's foundational architecture incorporates two key components driving the system's natural language understanding capabilities: GPT-2 language models and spaCy. GPT-2, a state-of-the-art language model, facilitates contextually rich response generation, ensuring that the Chatbot can engage in coherent and relevant conversations with users. Simultaneously, spaCy, a powerful natural language processing library, enables the extraction of essential keywords from user inputs, enhancing the system's ability to comprehend nuanced queries.

The user interface is crafted using Tkinter, providing an intuitive platform for users to interact seamlessly with the Chatbot. The integration of speech recognition capabilities extends the application's reach, allowing users to engage in voice-based conversations, thereby enriching the overall user experience. This inclusivity ensures accessibility for individuals with varying preferences and needs, contributing to a more inclusive digital environment.

Multilingual support is a pivotal aspect of the Chatbot's capabilities, facilitated by the incorporation of Google Translate. This integration broadens the application's reach across global audiences, enabling users to communicate in their preferred language and breaking down language barriers. The Chatbot's translation feature helps people from different cultures and backgrounds communicate with each other. This makes it easier for everyone to connect and share ideas online. The Chatbot also uses external tools like Google Search and YouTube to provide more information and entertainment. You can use the Chatbot to find information, learn new things, or watch videos without leaving the app.

The Chatbot is designed to help you in many ways. It can understand what you are saying and find the information you need. It can also learn from your feedback and improve over time. The Chatbot is always being updated to make it faster and more efficient. This means you can get the information you need quickly and easily.

A multi-agent system is proposed that has a chatbot as a middleware between the user and the outside world. This chatbot understands the user and their requirements. It is like having an assistant who understands the user & their requirements. The chatbot can make decisions on behalf of the user, thereby reducing the efforts of a user to carry out a task. Our concept of considering one task that is done with the help of a chatbot is demonstrated. The proposed system adapts and acts accordingly on behalf of the user.

The Chatbot is a great example of how technology can help people communicate and learn from each other. It shows how advanced language models and other tools can be used to create an intelligent and helpful assistant. The Chatbot is always improving, so it will be even more useful in the future. To improve the Chatbot's performance, it is important to monitor customer feedback and make necessary changes to the Chatbot's responses and behavior. Use the analytics and reporting tools available through the Chatbot platform to identify opportunities for enhancement and continually fine-tune the Chatbot's performance

ChatBot has the capability to recognize the specific domain of any query that is posted. The Cosine Similarity algorithm is applied here to find the right answer to the user query. The bot itself is intelligent enough to identify the frequently asked unanswered questions and notify the same for admin feedback. Once the admin provides a response for it, the bot is enhanced intelligent to look for new set of questions and answers and respond the same to the users in the future. This operation can be configurable in such a way that the system can decide the threshold limit for the unanswered questions according to the input.

An interactive tool that is designed to understand and respond to your messages or voice commands in a way that feels like chatting with a human. Our approach is to interact with users providing a more dynamic and reliable system. It should serve as a bridge between you and the digital world which truly understands your needs and has the ability to go beyond basic responses. This understanding empowers the chatbot to take initiative and make decisions on your behalf. It aims to measure tasks not just by completing them but by the ease and efficiency with which they are accomplished.

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**, Dean, School of Computer Science Engineering & Information Science, Presidency University for getting us permission to undergo the project.

We record our heartfelt gratitude to our beloved Associate Deans **Dr. Kalaiarasan C and Dr. Shakkeera L,** School of Computer Science Engineering & Information Science, Presidency University and **Dr. ZAFAR ALI KHAN ,** Head of the Department, School of Computer Science Engineering & Information Science, Presidency University for rendering timely help for the successful completion of this project.

We are greatly indebted to our guide **Dr. ASIF MOHAMMED H.B ,** Assistant Professor , School of Computer Science Engineering & Information Science, Presidency University for his 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 University Project-II Coordinators **Dr. Sanjeev P Kaulgud, Dr. Mrutyunjaya M S** and also the department Project Coordinators **Dr. Pallavi M**.

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

MAHESWAR REDDY

GANESH S

KARTHIK SHARMA

ARUN TEJA

LIST OF FIGURES

Sl. No.	Figure Name	Caption	Page No.
1	Figure 1.1	Architecture of text Chatbot	2
2	Figure 1.2	Speech recognition working	3
3	Figure 2.1	Bag of words of first 30 words	12
4	Figure 2.2	Bag of words of first random words	13
5	Figure 2.3	Tf score of first 30 words	14
6	Figure 2.4	Tf score of first random words	15
7	Figure 2.5	Idf score of first 30 words	16
8	Figure 2.6	Idf score of first random words	17
9	Figure 2.7	Tf-idf score of 30 words	18
10	Figure 2.8	Tf-idf score of random words	18
11	Figure 4.1	NLP	31
12	Figure 4.2	Flowchart of the code	35
13	Figure 6.1	Neural model	39
14	Figure 6.2	Basic chat deployment	43
15	Figure 7.1	Gantt chart	44
16	Figure 8.1	Accuracy of the model	45
17	Figure 8.2	Interface of the chatbot and search operation	45
18	Figure 8.3.1	Translation-from language	46
19	Figure 8.3.2	Translation-To language	46
20	Figure 8.3.2	Translated statement	47
21	Figure 8.4	End-operation	48
22	Figure b.1	Training model	55
23	Figure b.2	Interface	55
24	Figure b.3	Illustration of language translation and selecting from language	56
25	Figure b.4	Illustration of language translation and selecting To language	57
26	Figure b.5	Translated statement	58
27	Figure b.6	Termination	59

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT **ACKNOWLEDGMENT** **LIST OF FIGURES**	**v** **vii** **viii**
1.	**INTRODUCTION**	1
	1.1 DESCRIPTION	1
	1.2. TECHNOLOGY USED	2
	1.3 SPECIFICATIONS	2
	1.4 GOAL	4
2.	**LITERATURE REVIEW**	**5**
3.	**RESEARCH GAPS OF EXISTING METHODS**	**25**
	3.1 EMOTION AWARE SELF LEARNING CHAT-BOTS	25
	3.2 ASSESSING THE EMOTION ETHICAL IMPLICATIONS OF EMOTION AWARE SELF LEARNING CHAT BOT IN NLP	26
	3.3 USER EXPERIENCE AND INTERACTION	**27**
4.	**PROPOSED METHODOLOGY**	**31**
	4.1 NATURAL LANGUAGE PROCESSING	31
	4.2 DEEP LEARNING	33
	4.3 USER CENTRIC DESIGN AND EVALUATION.	34

5.	**OBJECTIVES**	**36**
6.	**SYSTEM DESIGN AND IMPLEMENTATION**	**40**
	6.1 SYSTEM SPECIFICATIONS	**40**
	6.2 IMPLEMENTATION	**41**
7.	**TIMELINE FOR EXECUTION OF PROJECT (GANTT CHART)**	**44**
8.	**OUTCOMES**	**45**
9.	**RESULTS AND DISCUSSIONS**	**49**
	9.1 GOOGLE SEARCH	49
	9.2 HANDS FREE ENVIRONMENT	50
	9.3 COMMUNICATION IS KEY CONVERSATIONAL SELF LEARNINNG BOTS	50
	9.4 NLP AND GPT-2	50
	9.5 SPEECH RECOGNITION ACCURACY	50
	9.6 WEB SEARCH AND EXTERNAL API	50
	9.7 FUTURE IMPROVEMENTS	51
	9.8 ETHICAL CONSIDERATIONS	51
10.	**CONCLUSION**	**52**
	10.1 CONCLUSION	52
	10.2 FUTURE SCOPE	53
	REFERENCES	**55**
	APPENDIX-A PSUEDO CODE	**57**
	APPENDIX-B SCREENSHOTS	**62**
	APPENDIX-C ENCLOSURES	**65**

CHAPTER-1
INTRODUCTION

As the present technology is improving and continues to advance day by day, self-learning chatbots are deemed to increase the productivity of the business they are deployed on and to increase the efficiency in responding to commands of the user effectively. These Systems hold the promise of further enriching effective interaction between humans and virtual humans which are machines that are developed to mimic human behavior. So far the Self learning bots in the present generation are intended to serve the customers in many ways more than the expected behavior. In addition to this, Present self-learning chatbots are often trained in Natural Language Processing and Natural Language understanding and incorporated with Natural language Generation capabilities allowing them to interpret the user inputs effectively and generate contextually appropriate responses.

Self-learning chatbots can accumulate knowledge and adapt to changing information. Unlike traditional rule-based systems, self-learning chatbots can analyze vast datasets, learn from user interactions, and autonomously update their internal models to enhance performance. This iterative learning process enables them to provide more accurate and personalized responses, creating a more natural and engaging user experience.

1.1 DESCRIPTION

The project “Self Learning Bot” aims to understand and respond to your messages or voice commands in a way that feels like chatting with a human. Our approach is to interact with users providing a more dynamic and reliable system. It should serve as a bridge between you and the digital world which truly understands your needs and has the ability to go beyond basic responses. This understanding empowers the chatbot to take initiative and make decisions on your behalf. It aims to measure tasks not just by completing them but by the ease and efficiency with which they are accomplished

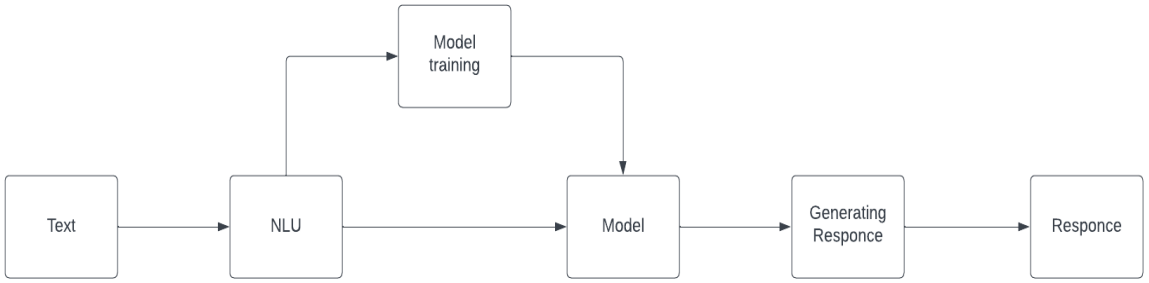


Figure 1.1 Architecture of text Chatbot

1.2 TECHNOLOGY USED

Hardware Requirements :

- Processor : Intel i3 or higher
- RAM : 4GB or higher
- STORAGE : 120GB or higher
- GPU enable

Software Requirements :

Operating System : Windows 7/8/10

Tools : Python , Anaconda

1.3 SPECIFICATIONS

1.3.1 GPT-2

The Chatbot is indeed built on the GPT-2 language model, which is developed by OpenAI.GPT-2 is a powerful language model that can generate coherent and relevant text that is contextual. By leveraging this cutting-edge language model, the Chatbot can understand and respond to user queries with a level of sophistication and nuance that was once considered unattainable. Yes, this particular GPT-2 is included in our Self-Learning Bot because it is necessary to respond to the user at least some idea about the input that is given, i.e if the input is out of context for our Self Learning Bot, It uses this GPT-2 pre-trained model to give minimal assistance to the user.

1.3.2 Tkinter

The Chatbot's graphical user interface (GUI) is indeed developed using Tkinter, which is a popular and versatile Python library.Tkinter is known for its simplicity and ease of use, making it an ideal choice for crafting an intuitive interface that ensures users,regardless of their technical proficiency, can seamlessly communicate with the Chatbot.

1.3.3 Speech Recognition

In addition to the GUI, the Chatbot also includes speech recognition capabilities, which further enhance accessibility. This feature offers users the option to engage in voice-based conversations, making it easier for them to interact with the Chatbot.

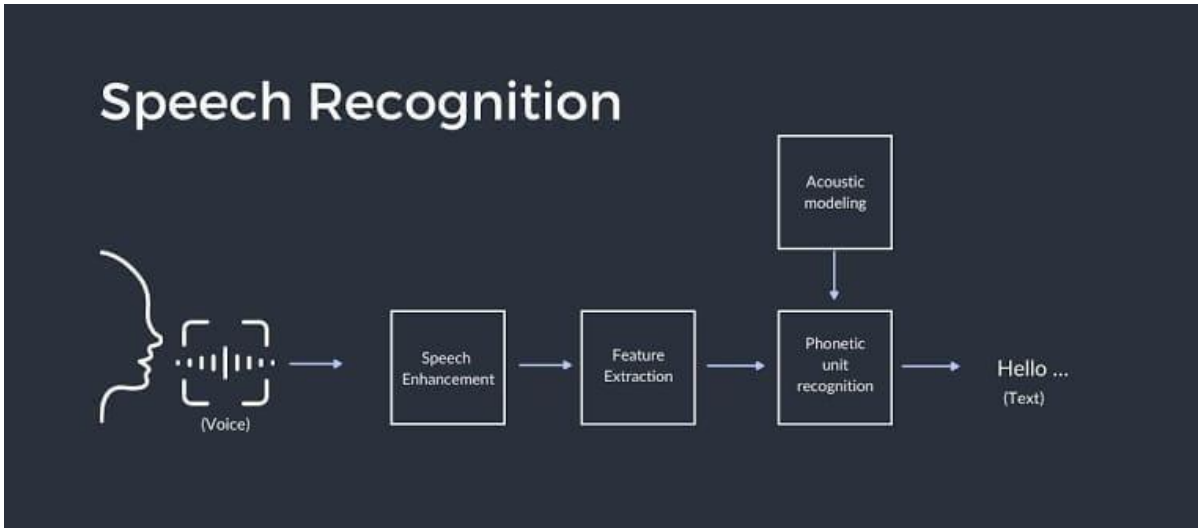


Figure 1.2 Speech recognition working

1.3.4 Translation

Language is essential when communicating with humans, it may become a problem if the user doesn't understand what the model replied to or just imagine if the user wants to communicate with a Self-learning chatbot in his native language or any language he knows. In our project, To overcome this problem, The Self Learning chatbot is integrated with Google Translate allowing users to interact with assistants in their preferred language. This not only facilitates

cross-cultural communication but also aligns to create a globalized and inclusive digital environment. This enhances user experience and positions the Chatbot as a dynamic and informative Assistant.

1.3.5 SpaCy

Spacy is a popular choice for self-learning bots because of its efficient tokenization, parsing, and named entity recognition capabilities. It offers pre-trained models and word embeddings for semantic understanding, which enhances the bot's contextual comprehension.With user-friendly APIs and comprehensive documentation, SpaCy makes it easy to implement natural language processing tasks. It's open-source nature and active community support ensure ongoing updates and collaborative development. Overall, SpaCy helps developers create smart and context-aware conversational agents quickly and accurately. This helps our Self Learning Bot to match the keywords exactly with the dataset or any spoken text by the user to provide accurate results in less time.

1.4 GOAL

This project also aims to provide a hands-free environment to the user by introducing voice commands and voice inputs so that the Self Learning bot recognizes the text transforms it into text and replies to you with the response in both text as well as in voice response. This particular task is achieved by importing the Speech Recognition model in Python and making the engine for speech recognition so that it activates and recognizes text every time the user speaks. The reason why it was been told is that it provides a hands-free environment compensated as the Self Learning Bot uses the voice command and performs several actions that are assigned to it based on the type of input it received. These tasks include playing any product review or any other video you are curious about on YouTube, Another task includes searching the web and providing direct navigation to the other pages about the topic you are looking for.

CHAPTER-2
LITERATURE SURVEY

2.1 Designing and Developing a Chatbot Using Machine Learning

IEEE.27 June 2019; *Praveen Kumar; Mayank Sharma; Seema Rawat; Tanupriya Choudhury*

2.1.1 OBSERVATIONS:

- Continuous acquisition and preprocessing of user data are essential for self-learning chatbots. Regularly updating and cleaning the dataset is crucial to ensure that the bot's learning is based on accurate and current information. By doing so, the chatbot can adapt to evolving language nuances and user preferences, enhancing its overall conversational capabilities.
- Incorporating advanced NLP techniques is crucial for the chatbot to effectively analyze and comprehend user input. This observation emphasizes the need for regular updates to the NLP models, enabling the bot to stay abreast of evolving language patterns. A well-integrated NLP system ensures the chatbot remains proficient in understanding user queries and providing contextually relevant responses.
- To foster self-improvement, the chatbot should incorporate a feedback loop mechanism. This enables users to provide feedback on the bot's responses, facilitating continuous refinement. Analyzing user feedback becomes a crucial aspect, allowing the system to identify areas for improvement and iteratively enhance its performance over time.
- A self-learning chatbot should be able to remember past interactions and retain context. This is achieved by implementing a memory mechanism that enables the bot to recall user history, preferences, and previous conversations. This ensures a more personalized and contextually relevant user experience, contributing to improved user satisfaction.

- To ensure that a self-learning chatbot can adapt and evolve with each interaction, it is essential to implement machine learning algorithms that support continuous learning.

2.2 A Self-Learning Chat-Bot From User Interactions and Preferences

IEEE.19 June 2020; *Thosani; Manas Sinkar; Jaydeep Vaghasiya; Radha Shankarmani*

2.2.1 OBSERVATIONS:

- Human-Centric Decision Making: The proposed chatbot, acting as a middleware in a multi-agent system, focuses on understanding the user and their needs. This goes beyond mere task completion and extends to the realm of decision-making on behalf of the user.
- Efficiency Through Understanding: The chatbot strives to not only interpret information provided by the user but also understand the underlying context and intent. This understanding forms the basis for efficient communication and task execution.
- Middleware Facilitation: By positioning the chatbot as a middleware, the system optimizes the flow of information between the user and the external environment. This architectural choice enables the chatbot to play a central role in facilitating communication.
- User Assistance and Empowerment: The chatbot, functioning as an assistant that truly comprehends the user, empowers users by handling tasks on their behalf.
- Reduction of Repetitive Interactions: Acknowledging the repetitive nature of many interactions with traditional chatbots, this system aims to break the monotony.
- Conceptual Shift in AI Interaction: The concept introduces a paradigm shift in AI interaction by moving beyond task-oriented conversations. Instead of merely completing tasks, the chatbot engages in a dynamic exchange that involves understanding, decision-making, and adaptive behavior.
- Task Delegation and Automation: With its decision-making capabilities, the chatbot is positioned to delegate and automate tasks based on user preferences.

This not only streamlines processes for the user but also introduces a level of trust in the chatbot's ability to act in the user's best interest. The system encourages users to rely on the chatbot as a capable and knowledgeable assistant.

- Intuitive Adaptability: The system demonstrates intuitive adaptability by learning from user interactions. It evolves its understanding of the user over time, adjusting its decision-making processes to align with changing preferences and requirements. This adaptability contributes to a more personalized and user-friendly experience, enhancing the overall effectiveness of the chatbot.

2.3 A Survey on Chatbot Implementation in Customer Service Industry through Deep Neural Networks

IEEE.30 December 2018; *Mohammad Nuruzzaman; Omar Khadeer Hussain*

2.3.1 OBSERVATIONS:

- Evolution from Rule-based to Neural Networks: The survey highlights the transition in chatbot development methodologies, moving from traditional hand-written rules and templates to the adoption of deep learning techniques, specifically Deep Neural Networks (DNNs). This shift signifies a move towards self-learning capabilities, allowing chatbots to adapt and improve their responses over time.
- End-to-end Neural Networks Dominance: The paper emphasizes the dominance of end-to-end neural networks in contemporary chatbot implementations. This approach enables a more holistic learning process, enabling the chatbot to grasp contextual nuances and generate responses that align better with human-like conversation
- Generative-based Models for Conversational Response: Deep Neural Networks are identified as powerful generative-based models, underscoring their effectiveness in addressing the challenges associated with conversational response generation. This indicates a trend towards leveraging generative models for more meaningful and contextually relevant interactions.
- Customer Service Challenges: The survey points out that approximately 75% of customers have encountered poor customer service. This highlights the significance of improving chatbot capabilities to generate meaningful, lengthy, and informative responses – a key area where self-learning bots can play a pivotal role in enhancing

overall customer experience.

- Rise of Natural Language Processing (NLP): The adoption of natural language

- processing techniques within chatbot implementations is evident from the survey. As deep learning models become more sophisticated, chatbots equipped with NLP capabilities can better understand and respond to user queries, contributing to more engaging and human-like interactions.
- Quality Conversation Impact: The paper explores how the limitations of current chatbot models impact the quality of conversation. Understanding the challenges in response generation sheds light on the importance of continual learning mechanisms within chatbots, emphasizing the need for self-improvement to bridge existing gaps in conversation quality.

2.4 Language translation of web-based content

IEEE.13 June 2013*; Bart Kahler; Brian Bacher; K. C. Jones*

2.4.1 OBSERVATIONS:

A self-learning bot, inspired by challenges in MT software, adapts to linguistic nuances, addressing issues like dialects and slang that may affect translation accuracy.Through iterative learning, the bot refines its language models to better understand and interpret colloquialisms, enhancing its ability to provide more contextually accurate translations.

- Contextual Correction Mechanism: Similar to the role of a native speaker in correcting translations, a self-learning bot employs contextual correction mechanisms.The bot learns to use cues from sentence structures and word usage, enabling it to capture the true meaning of phrases and sentences, resulting in more precise translations.
- Character Conversion Optimization: In response to challenges in character conversion errors, the self-learning bot optimizes its character conversion processes, especially when translating from Cyrillic, Asian, or Arabic languages to Western characters. This optimization minimizes errors in character association, mitigating the risk of changing meanings or forming incorrect word structures in the translated

text.

- Dynamic Error Resolution: The self-learning bot dynamically resolves errors introduced during character conversion, recognizing the dynamic nature of language and the need for real-time adjustments. By learning from its mistakes and iteratively improving, the bot enhances its accuracy in handling character conversion challenges, contributing to more reliable translations.
- Integration of Open Source MT: The bot integrates open-source MT, showcasing its ability to leverage existing solutions and incorporate external knowledge. This integration allows the bot to benefit from collective improvements in MT algorithms and models, contributing to a more robust translation system.
- Utilization of ISO Character Mapping: Drawing from the authors' solution, the self-learning bot incorporates ISO character mapping as a fundamental part of its translation process.This utilization ensures proper character conversion, aligning with international standards and significantly improving the accuracy of translations for web-based content.
- Continuous Learning from Multilingual Data:
- The self-learning bot continuously learns from multilingual datasets to enhance its language understanding and translation capabilities.
- Exposure to diverse linguistic data aids the bot in recognizing patterns, idioms, and cultural nuances, allowing it to produce more culturally sensitive and accurate translations.
- Proactive Identification of Translation Challenges:
- Through its self-learning capabilities, the bot becomes proactive in identifying and addressing specific challenges in translation, such as dialects and character conversion errors.
- This proactive approach contributes to the bot's agility in handling a variety of linguistic complexities encountered in web-based content.
- Evolutionary Response to Web-Based Linguistic Trends:
- The self-learning bot evolves in response to changing linguistic trends in web-based content.
- By staying abreast of emerging linguistic patterns and expressions, the bot ensures its

translations remain contemporary and aligned with the evolving nature of online language usage.

2.5 Incorporating Pre-trained Model into Neural Machine Translation

IEEE.28 June 2021; Jiaxing Song; Weidong Liu

2.5.1 OBSERVATIONS:

- 1. Language Proficiency Enhancement: Highlighting how pre-trained models elevate the language skills of chatbots, enabling them to engage in more sophisticated conversations with a higher level of proficiency.
- 2. Learning Efficiency: Emphasizing the accelerated learning curve and reduced dependence on extensive training datasets, showcasing the efficiency gains in the development and deployment of language translation capabilities.
- 3. Linguistic Sensitivity: Discussing the chatbots' increased sensitivity to linguistic nuances, such as idioms and cultural context, showcasing their ability to provide more accurate and culturally aware translations.
- 4. Autonomous Skill Refinement: Describing the autonomous learning capability of chatbots, allowing them to refine their language translation skills over time through real-world interactions.
- 5. Multilingual Versatility: Exploring how pre-trained models enable chatbots to seamlessly navigate and translate between different languages, showcasing their versatility in multilingual scenarios.
- 6. Resource Efficiency: Illustrating how the integration of pre-trained models optimizes resource utilization, making chatbots more efficient in processing linguistic data and providing quicker responses.
- 7. Cross-Language Communication: Highlighting the streamlined communication between different languages facilitated by chatbots, enhancing the overall user experience and breaking down language barriers effectively.
- 8. Reduced Dependency on Extensive Data: Discussed how pre-trained models reduce the reliance on large training datasets, leading to quicker development cycles and the ability of chatbots to operate effectively with smaller datasets.

- 9. Continuous Learning for Accuracy: Exploring the concept of continuous learning, where chatbots improve their language translation accuracy over time by adapting to evolving language patterns.

- 10. User-Centric Adaptation: Emphasizing the user-centric approach in communication, where chatbots adapt their responses to user preferences, creating a personalized and engaging experience that goes beyond basic language translation.

2.6 Modified TF-IDF Term Weighting Strategies for Text Categorization

IEEE.02 June 2021; *Ghulam Musa Raza; Zainab Saeed Butt; Seemab Latif; Abdul Wahid*

2.6.1 OBSERVATION

TF-IDF vectorization:

To classify our data into number-based vectors, and natural language processing techniques for feature extraction we use a count vectorizer and tf-idf has been used to get a bag of words from the data.tf-idf used a count vectorizer to count the term frequencies (tf) and also inverse document

- Frequency(idf) which help evaluate the measure of word relevancy appearing in multiple documents. One dataset is proposed by the team and another one is pre trained model called the gpt2 model to get our input into machine learning format we have converted the data into vectors that represent the words into numbers by using feature extraction techniques of NLP like count vectorizer and tf-idf. It is useful to assign relevance scores for potential answers.
- TF-IDF constitutes two terms: Term Frequency (TF) and Inverse Document Frequency (IDF). ‘TF’ tells us that if the term ‘t’ is very frequent in a document ‘d’ then t is very important
- Similarly, ‘IDF’ measures the rarity of ‘t’ with respect to the entire collection i.e. terms that appear in many documents are less informative. The number of documents

that contain the term t in the entire collection gives its document frequency or global frequency.

- There are three modified tf-idf techniques are proposed which improve the text classification process:
- TF-IDF based on modified inverse document frequency: Consider those words that are very frequent in documents and also in the entire collection (except stop-words). This helps to increase the discriminative power of traditional IDF.
- Weight(i, j) = W2ij = T F(i, j) ModifiedIDF (i)
- TF-IDF based on class frequency: The class frequency helps to identify whether a term is relevant to a particular class or not. It gives weight to the terms in class characteristics. In text classification, adding weights to the terms based on their class frequency will improve the classification accuracy.
- TF-IDF based on normalized length: By adding the length normalizing factor to TF-IDF will increase the importance of TF and reduces the weights of the terms which are less frequent but have relatively higher TF-weighting in the document.
- Term Frequency (TF):Calculating the frequency of each terms with in a document. It's often calculated as the number of times a term appears in a document divided by the total number of terms in that document:

TF(t,d) = Number of times t appears in document d/Total number of documents in d

Inverse Document Frequency (IDF): Assessing the importance of a term across the entire corpus. It's computed as the logarithm of the ratio number of documents to the number of documents containing the term, preventing the overemphasis of common terms:
IDF(t,D) = log(Total number of documents in the corpus/Number of documents containing term t)

Words/Documents	Counts	Tags
What	126	391
Data	304	
Explain	79	
How	123	
Describe	36	
Sql	16	

Fig 2.1 word count of first 30 tags

TF-IDF Calculation: Combining TF and IDF to get the TF-IDF score for each term in a document. The TF-IDF score for a term in a document is calculated as:

TF-IDF(t,d,D) = TF(t,d) x IDF(t,D)

Words/Documents	Counts	Tags
Are	228	391
Significance	10	
Describe	36	
Concept	48	
Tradeoff	7	
Machine	177	

Fig 2.2 word count of random 30 tags

TF(t,d) = Number of times t appears in document d/Total number of documents in d

TF-IDF	What	Is	Explain	How	Describe	Sql
tag1	0.1036	0.1036	0	0	0	0
tag2	0	0	0.076	0	0	0
tag3	0.099	0	0	0	0	0
tag4	0	0	0	0.125	0	0
tag5	0.073	0.073	0	0	0	0
tag6	0	0	0	0	0	0
tag7	0.057	0.057	0	0	0	0
tag8	0	0	0.115	0	0	0
tag9	0	0	0	0	0	0.2
tag10	0	0	0	0	0.1	0

Fig 2.3 tf score of first 30 tags

TF-Score

Words/Documents	Are	Significance	Describe	Concept	Tradeoff	Machi
tag11	0	0.1	0	0	0	0.1
tag12	0	0	0.125	0	0.125	0.125
tag13	0	0	0	0	0	0
tag14	0.1	0	0	0	0	0
tag15	0	0	0	0.111	0	0.111
tag16	0	0	0	0	0	0
tag17	0	0	0	0	0	0
tag18	0	0	0	0	0	0
tag19	0.1	0	0	0	0	0
tag20	0	0	0	0.166	0	0

Fig 2.4 tf score of random 30 yagss

IDF(t,D) = log(Total number of documents in the corpus/Number of documents containing term t)

IDF-Score

Words/Documents	IDF-Value
What	0.518
Is	0.518
Explain	0.698
How	1
Describe	1
Sql	1

Fig 2.5 idf score of first 30 tags

IDF-Score

Words/Documents	IDF-Value
Are	0.698
Significance	1
Describe	1
Concept	0.698
Tradeoff	1
Machine	0.522

Fig 2.6 idf score of random 30 tags

TF-IDF Calculation: Combining TF and IDF to get the TF-IDF score for each term in a document. The TF-IDF score for a term in a document is calculated as:

TF-IDF(t,d,D) = TF(t,d) x IDF(t,D)

TF-IDF	What	Is	Explain	How	Describe	Sql
tag1	0.1036	0.1036	0	0	0	0
tag2	0	0	0.076	0	0	0
tag3	0.099	0	0	0	0	0
tag4	0	0	0	0.125	0	0
tag5	0.073	0.073	0	0	0	0
tag6	0	0	0	0	0	0
tag7	0.057	0.057	0	0	0	0
tag8	0	0	0.115	0	0	0
tag9	0	0	0	0	0	0.2
tag10	0	0	0	0	0.1	0

Fig 2.7 tf-idf score of first 30 tags

TF-IDF	Are	Significance	Describe	Concept	Tradeoff	Machine
tag11	0	0.1	0	0	0	0.052
tag12	0	0	0.125	0	0.125	0.065
tag13	0	0	0	0	0	0
tag14	0.069	0	0	0	0	0
tag15	0	0	0	0.077	0	0.057
tag16	0	0	0	0	0	0
tag17	0	0	0	0	0	0
tag18	0	0	0	0	0	0
tag19	0.069	0	0	0	0	0
tag20	0	0	0	0.115	0	0

Fig 2.8 tf-idf score of random 30 tags

2.7 Overview of the Speech Recognition Technology

IEEE.13 September 2012; *Jianliang Meng; Junwei Zhang; Haoquan Zhao*

2.7.1 OBSERVATION:

Speech Recognition is the machine on the statement or command of human speech to identify understand and react accordingly. It allows the bot to automatically identify and understand human spoken language through signal processing and pattern recognition the voice through the microphone is converted into an electrical signal.

Representative speech recognition methods include dynamic time warping (DTW), hidden Markov model (HMM), vector quantization (VQ), artificial neural network (ANN), support vector machine (SVM), and so on. The article focuses on two methods hidden Markov model (HMM) and artificial neural network (ANN).

- The HMM model parameters represent the time-varying characteristics of the voice signal. It uses statistical methods of training the underlying acoustic model and the upper voice model into
- the unified voice recognition search algorithm can obtain better recognition results and can be used for continuous speech recognition, but the drawback is the need to be very
- sophisticated calculations and a longer training sequence.
- Application of speech recognition technology:
- There are many speech recognition applications like the system-VRCP system developed by AT&T in 1992. In September 1996, Charles Schwab launched the first large-scale commercial speech recognition application system the stock quotation system. Another development branch of speech recognition technology is the development of the telephone voice recognition technology.
- This will be able to telephone inquiries, automatic writing as well as some specialized operations. it can provide 24/7 phone banking service. There are many problems faced by speech recognition and it is mainly reflected in the dependence on the environment
- If the environment is noisy then it is very difficult to progress the speech like voice, slow speech rate, etc.

2.8 High Accuracy Conversational AI Chatbot Using Deep Recurrent Neural Networks Based BiLSTM Model

IEEE.29 January 2021; *Prasnurzaki Anki; Alhadi Bustamam; Herley Shaori Al-Ash; Devvi Sarwinda*

2.8.1 OBSERVATIONS:

- Model Selection: The decision to employ the BiLSTM model is grounded in its efficacy in handling sequential data. This bidirectional Long Short-Term Memory (BiLSTM) architecture is well-suited for capturing dependencies in both preceding and succeeding words, offering a comprehensive understanding of context in a conversation.
- Greedy Method Implementation: The implementation strategy involves the utilization of the greedy method. This method is adopted to enhance data processing speed during program execution.
- Program Evaluation Method: To validate the effectiveness of the chatbot, a robust program evaluation method is essential. This method is designed to verify whether the output generated by the chatbot aligns with the user's expectations. It serves as a quality assurance step to ensure the chatbot's responses meet the desired criteria.
- Self-Learning Bot: The self-learning aspect of the chatbot involves continuous improvement based on user interactions. As users engage with the chatbot, it learns from the data collected during question-and-answer sessions. This iterative learning process enhances the chatbot's knowledge over time, enabling it to provide more accurate and contextually relevant responses.
- Optimizing Results: The primary goal of the chatbot is to optimize results based on highly rated questions asked in a service center. By leveraging the capabilities of the BiLSTM model, the chatbot aims to provide precise and relevant information, enhancing the overall efficiency and effectiveness of the service center operations.
- Contextual Understanding: The bidirectional nature of the BiLSTM model enables the chatbot to grasp the context of user queries more comprehensively. This contextual understanding plays a pivotal role in generating responses that not only answer specific questions but also consider the broader conversation flow.
- Enhanced User Experience: By integrating self-learning mechanisms and advanced models like BiLSTM, the chatbot aims to provide an enhanced user experience.

2.9 Extractive Automatic Text Summarization using SpaCy in Python & NLP

IEEE.20 April 2021; *SWARANJALI JUGRAN; ASHISH KUMAR; BHUPENDRA SINGH TYAGI; VIVEK ANAND*

2.9.1 OBSERVATIONS :

- Adaptability and Continuous Learning:Self-learning chatbots are designed to adapt and improve over time by learning from user interactions and feedback. Continuous learning allows the chatbot to stay relevant and provide more accurate and personalized responses as it gathers more data.
- The ability of a self-learning chatbot to engage users effectively is crucial for its success. Natural Language Processing (NLP) capabilities contribute to understanding and responding to user queries in a human-like manner.
- Data Privacy and Security:Handling sensitive information requires robust security measures to ensure user data privacy. Self-learning chatbots should adhere to data protection regulations and implement encryption and secure communication protocols.
- Context Awareness:A successful self-learning chatbot should understand and maintain context throughout a conversation. Context awareness enables the chatbot to provide coherent and relevant responses, creating a more natural and seamless interaction.
- Multimodal Capabilities:Integrating multimodal capabilities, such as understanding both text and images, can enhance the chatbot's ability to comprehend and respond to diverse user inputs.
- Proactive Engagement:A successful self-learning chatbot can initiate conversations, ask follow-up questions, and proactively engage users, creating a more interactive and user-friendly experience.
- Fallback Mechanisms:Implementing effective fallback mechanisms is important for scenarios where the chatbot encounters queries or situations outside its training scope. Providing a graceful exit or redirection is crucial.
- Explainability and Transparency:Ensuring that the chatbot can explain its decisions and reasoning to users enhances transparency. Users should understand why the chatbot provided a particular response.

- User Intent Prediction:Accurately predicting user intent is a key aspect. Self-learning chatbots should be proficient in understanding the underlying goals or requests behind user queries to generate appropriate responses.

2.10 Text Generation and Prediction System: Pre-training on New Corpora Using BERT and GPT-2

IEEE.30 July 2020*; Yuanbin Qu; Peihan Liu; Wei Song; Lizhen Liu; Miaomiao Cheng*

2.10.1 OBSERVATION:

- To generate text there are many methods like Recurrent neural network(RNN) or Long-Short term memory(LSTM) the encode-decode method and the sequence-to-sequence method .It converts the sentence as a fixed vector first, and then uses a vector to generate the sentence
  there are some language models like gpt2 and bert for
- Text generation. the Open-AI gpt2 model is built using a transformer module, which is
  decoder-only module.
- The model takes tokens as input and generates outputs only one token at a time. after each token is generated, the token is added to the previously generated token sequence
- while using the model, the main task is to generate a sentence based on the keyword. the main idea of the model is to generate the answer in a loop based on the input sequence of known text
- GPT-2 is a Transformer architecture that was notable for its size (1.5 billion parameters) on its release. The model is pre-trained on a WebText dataset - text from 45 million website
  links.
- It largely follows the previous GPT architecture with some modifications:
- Layer normalization is moved to the input of each sub-block, similar to a pre-activation residual network and an additional layer normalization was added after the final self-attention block.

- A modified initialization which accounts for the accumulation on the residual path with model depth is used. Weights of residual layers are scaled at initialization by a factor of
  where is the number of residual layers.
- The vocabulary is expanded to 50,257. The context size is expanded from 512 to 1024 tokens and a larger batch size of 512 is used in the code, the bot first checks in the JSON file if the user input is not present in the data file then it is navigated into the gpt2 model it will take 30-45 seconds to load the data and check in the vocabulary that consists of fifty thousand.

CHAPTER-3
RESEARCH GAPS OF EXISTING METHODS

Self Learning Bot may vary from each other in different aspects as per their aim of creation and deployment but we can provide a brief overview of common approaches and technologies that can be used in the development of Self Learning Bot .

3.1 EMOTION AWARE SELF LEARNING CHATBOTS

Self-learning bots have been designed to understand user intent and context, and provide relevant responses. However, these bots often lack the ability to accurately perceive and respond to user emotions, which play a crucial role in natural human communication. Emotion-aware bots have the potential to enhance user experience, improve engagement, and tailor responses based on the user's emotional state.

3.1.1 Emotion recognition

Self-learning bots are designed to understand user intent and context, and provide relevant responses. However, these bots often lack the ability to accurately perceive and respond to user emotions, which play a crucial role in natural human communication. Emotion-aware bots have the potential to enhance user experience, improve engagement, and tailor responses based on the user's emotional state.

3.1.2 Adaption to Emotions

This requires developing NLP models that can accurately recognize and interpret emotions expressed in textual inputs, which is a significant challenge. Textual cues, such as sarcasm or subtle emotional nuances, are often difficult for existing models to grasp. Addressing ethical concerns related to emotion detection, ensuring user consent, and managing data privacy become critical factors.

3.1.3 Long-term Emotional Context Understanding

Most existing self-learning bots focus on short-term contextual understanding. The challenge lies in building models that can capture the understanding of the user's emotional content in broader contexts, improving the overall relationship between user and Self Learning bot.

3.1.4 Contextual Adaptation to Emotions

Integrating emotional-cognitive skills into the dialogue process is complex. Self Learning Bots need to dynamically adjust their responses based on the user's emotional state while maintaining ongoing communication within the conversation.

In small Conclusion , The research gap in emotion-aware self-learning bots has broad implications in a variety of applications including customer service, mental health support, and both human-computer interaction Such Self Learning bots enable users greater role satisfaction, improved support.

3.2 ASSESSING THE EMOTION ETHICAL IMPLICATIONS OF EMOTION AWARE SELF LEARNING CHAT BOT IN NLP

While incorporating emotional intelligence into self-learning bots has the potential to enhance the user experience, there is a notable lack of comprehensive research on the ethical implications associated with emotion a discovery and modification of Research gaps in the use of Self Learning bots for learning emotional intelligence It lies in the need for a deeper understanding of ethical considerations and potential risks involved

3.2.1 Informed Consent and User Awareness

It is essential to ensure that users are absolutely knowledgeable and privy to the extent to which their emotional information is being collected and applied to tailor Self Learning bot responses. Obtaining informed consent from customers is a essential step in this method. Consent must be unfastened, precise, informed, unconditional, and unambiguous with a clear affirmative action signifying the user's assent to the processing of private statistics to the

quantity essential for a designated motive.

3.2.2 Bias and Fairness in Emotional Analysis

The potential for biases in emotion reputation algorithms poses moral demanding situations. Models skilled on positive datasets might also show off biases that may impact consumer studies disproportionately. Addressing those biases and ensuring equity in emotional analysis is a vital but understudied factor.

3.2.3 User Autonomy and Emotional Transparency

Emotion-conscious bots have the capacity to steer person feelings intentionally or by chance. The loss of clear hints and safeguards increases issues approximately the ethical use of emotional records and the potential for manipulation in various contexts, together with mental health help or advertising.

3.2.4 Privacy Risks and Emotional Data Security

Emotional statistics is touchy, and there is a lack of comprehensive studies on a way to securely take care of and keep this records. Ensuring the privacy of users' emotional states is a huge challenge that needs to be addressed to prevent unauthorized get entry to and ability misuse.

In small Conclusion , The research gap neglecting the moral issues associated with emotion-conscious self-learning bots can lead to unintentional consequences, erode consumer trust, and probably infringe on person privacy. It is essential to evaluate the broader societal implications and ensure accountable improvement and deployment of these technology.

3.3 User Experience and Interaction

Evaluating the user enjoy of the Self Learning Chatbot interface and explore different methods to beautify consumer interplay the use of the existing tried approach which amassed handiest constrained actions . Investigating consumer preferences for conversational sellers

and adapt the interface thus. This might not be efficiently carried due to the scope of the library this is getting used to develop the interface.

3.4 Speech Recognition Accuracy

Assess the accuracy of the speech recognition system, in particular in noisy environments. The Accuracy performs key position in recognition manner this is performed with the aid of the version to transform the speech into textual content , So the Accuracy is probably affected when we disclose the model to noisy environments . Exploring advanced speech popularity models or techniques to improve accuracy that could boom the accuracy of the model.

3.5 Personalization and Context Understanding

No approach promises the right expertise of the context provided as the Self Learning Bot constantly tries to improve itself. • In the technique of enhancing it can attempt to show to new facts and new complications are raised accordingly • Enhancement of expertise of context inside a communication is inevitable to carry out and method the enter.

3.6 Dynamic Learning and Adaptability

Explore mechanisms for dynamic studying and variation of the chatbot over the years leads to new approaches of processing the input furnished by person. Implement techniques for the chatbot to examine from user comments and improve its responses can be complicated.

3.6.1 Integration of Multimodal Inputs

Consider incorporating multimodal inputs, inclusive of pictures or video, to decorate the chatbot's knowledge and responses. • Explore how the chatbot can cope with both textual and non-textual consumer inputs seamlessly.

3.6.2 Fine-tuning and Training Strategies

Investigate extra efficient and effective strategies for excellent tuning or training the language model. Explore transfer studying techniques for adapting pre-trained models to

unique domain names.

3.7 Exploring Domain-specific Adaptation Challenges

3.7.1 Domain-particular Lexical and Syntactic Variability

Different domains frequently have particular vocabularies, expressions, and syntactic systems. Research is wanted to recognize how emotion-aware bots can adapt their fashions to correctly apprehend and respond to domain-specific emotional cues.

3.7.2 Specialized Emotional Triggers

Various domains may also have awesome emotional triggers and sensitivities. Investigating how emotion-conscious bots can discover and accurately respond to feelings relevant to particular contexts, inclusive of healthcare, customer service, or training, is vital for their effectiveness.

3.7.3 User Expectations and Communication Styles

Users in special domain names may have awesome expectancies and conversation patterns. Understanding how emotion-aware bots can align with those expectations and adapt their emotional responses to healthy the communication norms of unique domain names is a pressing research challenge.

3.7.4 Multi-modal Context Integration

Many domains involve multi-modal interactions, consisting of text, photos, and probably audio. Research is needed to explore how emotion-aware bots can effectively integrate and reply to emotional cues across a couple of modalities, making sure a holistic know-how of person feelings.

3.7.5 Dynamic Environmental Contexts

Different domain names can also have dynamic environmental factors influencing person emotions. Research ought to consciousness on growing fashions able to adapting to changing contextual factors and appropriately reflecting the emotional states of customers in response to various environmental situations.

Neglecting area-precise adaptation demanding situations can limit the effectiveness of emotion-aware self-getting to know bots in actual-global applications. Understanding and addressing these demanding situations are vital for the a hit deployment of emotion-conscious structures in diverse domains, ensuring that they resonate with users and provide meaningful emotional interactions.

3.8 Handling Ambiguity and Uncertainty

Enhance the chatbot's potential to address ambiguous queries or situations with a couple of interpretations. Investigate methods to explicit uncertainty within the chatbot's responses while suitable. Assess and address capability biases inside the language model's responses. Explore methods to make sure moral behavior and mitigate accidental consequences in consumer interactions.

3.9 Real-time Learning and Feedback Loop

Implement a real-time gaining knowledge of mechanism that permits the chatbot to constantly improve in the course of interactions. Establish a feedback loop for users to provide feedback on the chatbot's responses.

3.10 Integration with External Knowledge Bases and Performance Optimization

Explore techniques for integrating the chatbot with external understanding bases to enhance its information and facts retrieval competencies. Investigate dynamic updates from dependable resources to preserve the chatbot's knowledge up-to-date.

Assess the scalability of the chatbot system, particularly in scenarios with a big range of concurrent users. Optimize the performance of the device, considering computational resources and response times.

CHAPTER-4
PROPOSED METHODOLOGY

4.1 Natural Language Processing :

NLP is a way for computers to understand and use human language, both written and spoken. It helps computers break down text into smaller parts, and figure out what they mean, and what the speaker or writer wants to say. Computers learn from a lot of text data how to do things like finding out how people feel, translating between languages, and creating new text. NLP makes possible many things we use every day, like chatbots, virtual assistants, search engines, language translation, sentiment analysis, text summarization, and more. NLP is always improving with new methods from deep learning, making computers more capable of understanding language as humans do.

4.1.1 Tokenization:

Tokenization is the manner of breaking down a text into person devices, usually words or sub-words. It serves as the first step in analyzing and processing textual records.

4.1.2 Stemming:

Stemming entails reducing phrases to their base or root form, removing suffixes or prefixes. This allows in taking pictures the center meaning of words and simplifying analysis.

4.1.3 Lemmatization:

Lemmatization is just like stemming but specializes in decreasing words to their base or dictionary shape (lemma) to normalize variations. It ensures that distinctive inflections of a word map to the equal root.

4.1.4 Intent Recognition

Intent reputation identifies the purpose or cause at the back of a user's enter. It permits the bot to apprehend what the user wants and reply as a result, contributing to powerful interplay.

4.1.5 Text Summarization

Language technology includes growing human-like text responses. In the context of self-studying bots, it is important for generating coherent and contextually applicable responses in conversations with users. These sub-domain names collectively make a contribution to the development of sophisticated NLP systems, enabling self-getting to know bots to recognize and generate human-like language in numerous contexts. They shape the inspiration for building wise conversational agents that continuously enhance via gaining knowledge of from user interactions.

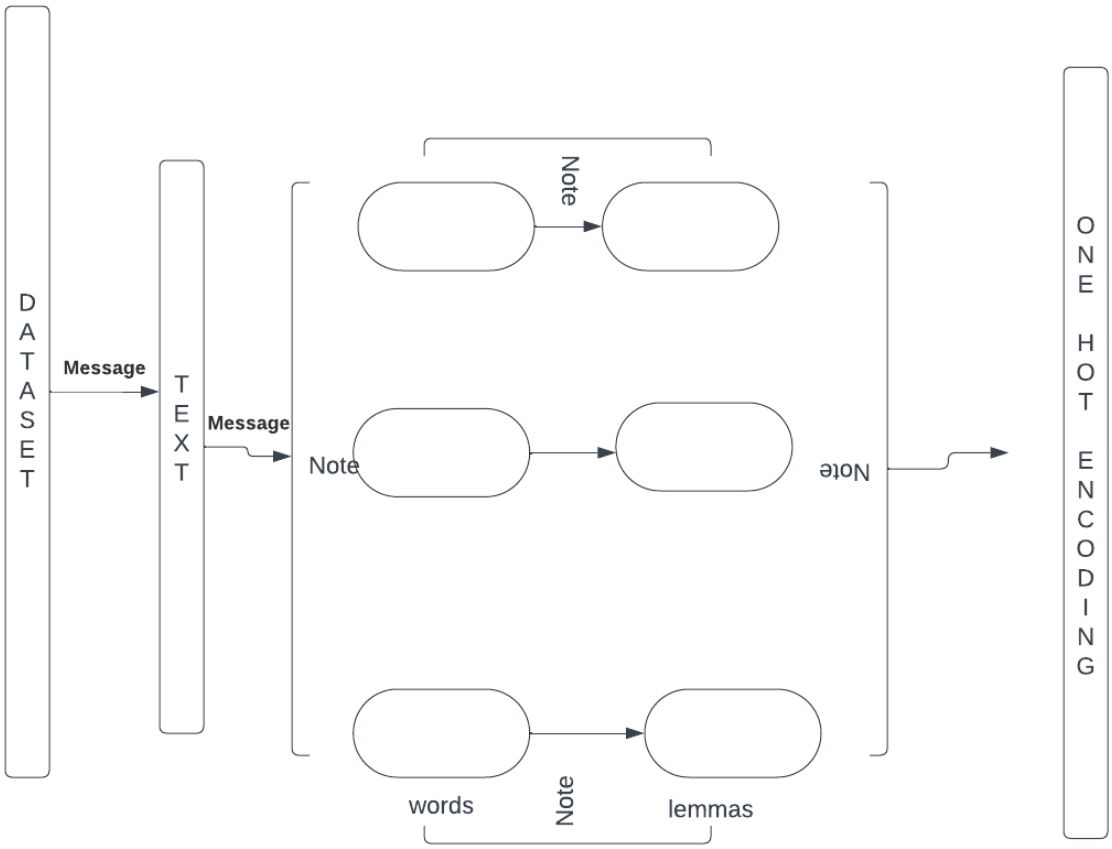


Fig 4.1 NLP

4.2 Deep Learning

Deep learning is a type of machine learning that uses artificial brain-like networks to learn from data. It can find hidden patterns and features that are hard to see otherwise. Deep learning networks have many layers that change the input data into more and more useful information. To train these networks, we need to tweak millions of settings until they can make good predictions. Some common types of deep learning networks are CNNs for pictures and RNNs for words or sounds. Deep learning has done amazing things in making computers see, understand, and talk. But it also has some problems, like needing a lot of data, power, and explanation.

4.2.1 Recurrent Neural Networks (RNNs):

RNNs are a form of neural community architecture designed to work with sequential records, making them appropriate for tasks involving sequences of phrases or characters. They have been utilized in chat bot development to keep context throughout one-of-a-kind turns in a verbal exchange

4.2.2 Transformer Architecture:

Transformers are a type of neural community architecture that relies on self-attention mechanisms to procedure enter information in parallel. The Transformer architecture has been pivotal in natural language processing tasks, including chat bot development contextual records

4.2.3 Sequence-to-Sequence (Seq2Seq) Models :

Seq2Seq models use an encoder-decoder architecture, in which the encoder strategies the input collection, and the decoder generates the output series. These fashions were employed in chat bots for obligations like language translation and speak generation.

4.2.4 GPT (Generative Pre-skilled Transformer) Models:

GPT models, which includes GPT-3, are massive-scale language models pre-educated on numerous datasets. They can be high-quality-tuned for specific obligations, which include chat bot responses, and are recognized for their ability to generate coherent and contextually relevant text.

4.3 User-Centric Design and Evaluation:

User Research: Conduct surveys, interviews, or usability exams to understand user preferences, expectations, and pain points when interacting with the chatbot.

Iterative Design: Implement iterative layout cycles to contain consumer remarks into the interface, making sure a person-friendly revel in.

4.4Enhancing Speech Recognition and Natural Language Processing (NLP):

4.4.1 Advanced Speech Recognition: Explore superior speech recognition models or preprocessing strategies to decorate accuracy, specially in noisy environments.

4.4.2 Context Understanding: Implement context-recognition in the NLP model to understand conversational context and person intents extra as it should be.

4.5 . Dynamic Learning and Adaptation:

4.5.1 Continuous Learning: Develop mechanisms for the chatbot to research continuously from user interactions, updating its expertise and responses.

4.5.2 Adaptive Responses: Implement algorithms that adapt the chatbot's responses based on historical person interactions and comments. Four. Multimodal Integration and Personalization.

4.5.3 Multimodal Inputs: Integrate support for multimedia inputs (photos, movies) to enrich person interactions and provide extra comprehensive responses.

4.5.4 Personalization Strategies: Explore strategies to customize the chatbot's responses based on user records, possibilities, and contextual cues.

4.6. Ethical Considerations and Bias Mitigation:

4.6.1 Bias Detection and Mitigation: Develop strategies to come across and mitigate biases in the language version's responses, ensuring equity and inclusivity.

4.6.2 Ethical Guidelines: Establish clear ethical tips for the chatbot's behavior and responses, prioritizing consumer privateness and agree with.

4.7. Real-time Learning and Feedback Loop:

4.7.1 Feedback Integration: Create a feedback loop wherein customers can provide comments on the chatbot's responses, and the system contains this feedback for improvement in actual-time.

4.7.2 Active Learning: Implement active getting to know strategies to pick out areas where the chatbot lacks information and actively seeks clarification or additional

statistics.

4.8. Integration with External Knowledge Bases:

4.8.1 Knowledge Base Integration: Explore methods to integrate external, reliable knowledge sources to beautify the chatbot's understanding and statistics retrieval capabilities.

4.8.2 Dynamic Updates: Develop mechanisms to maintain the chatbot's expertise base up-to date via retrieving and assimilating records from tested resources.

4.9. Security, Privacy, and Scalability:

4.9.1 Security Measures: Implement sturdy security protocols to ensure stable interactions and defend consumer information.

4.10 Cultural and Language Adaptability:

4.10.1 Multilingual Support: Expand language guide and cultural adaptability to cater todiverse person demographics and linguistic contexts.

4.10.2 Cultural Sensitivity: Ensure the chatbot's responses are culturally sensitive and contextually suitable for extraordinary consumer organizations.

4.11 . Evaluation and Validation:

4.11.1 Quantitative Metrics: Define performance metrics (accuracy, consumer pride, reaction time) to evaluate the effectiveness of every enhancement.

4.11.2 User Testing: Conduct rigorous testing and evaluations with real users to validate improvements and iterate primarily based on comments.

4.12. Documentation and Transparency:

4.12.1 Documentation: Maintain comprehensive documentation of methodologies, model updates, and enhancements for transparency and reproducibility.

4.12.2 User Education: Educate users approximately the chatbot's skills, barriers, and ethical recommendations for a obvious consumer-bot interaction.

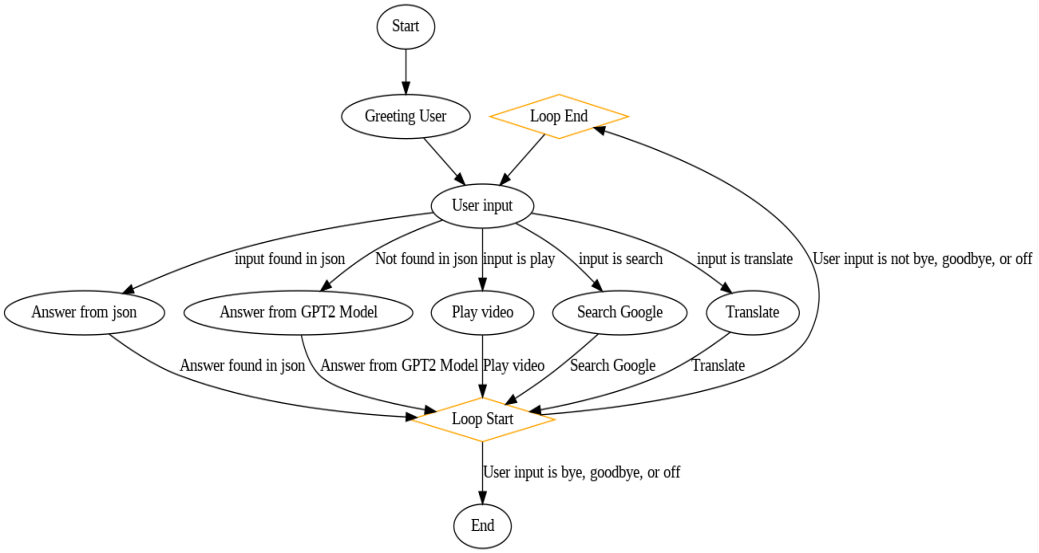


Figure 4.2 flowchart of the code

4.13. Continuous Improvement and Future Research:

4.13.1 Iterative Refinement: Embrace an iterative approach for non-stop improvement, incorporating new studies findings and technological improvements.

4.13.2 Future Research Directions: Identify ability destiny research guidelines based totally on going technological trends and user needs

CHAPTER-5

OBJECTIVES

5.1. Enhance User Interaction:

Objective: Improve the overall user experience by implementing a user-friendly interface and intuitive interactions.

Key Results:

- Increased user satisfaction ratings.
- Reduced user dropout rates during interactions.

5.2. Improve Speech Recognition Accuracy:

Objective: Enhance the accuracy of the speech recognition system for better understanding of user voice commands.

Key Results:

- Increased accuracy in transcribing spoken language.
- Reduced instances of misinterpretation or errors.

5.3. Optimize Natural Language Processing (NLP):

Objective: Optimize the NLP model to better understand user intents and provide context-aware responses.

Key Results:

- Improved accuracy in identifying user intents.
- More contextually relevant responses.

5.4. Enable Dynamic Learning and Adaptation:

Objective: Implement mechanisms for the chatbot to learn continuously and adapt its responses based on user interactions.

Key Results:

- Increased knowledge retention over time.
- Improved responsiveness to user preferences.

5.5. Integrate Multimodal Inputs:

Objective: Enable the chatbot to process and respond to multimedia inputs, enhancing the range of user interactions.

Key Results:

- Successful processing of image or video inputs.
- Improved user engagement with multimedia capabilities.

5.6. Address Ethical Considerations and Bias:

Objective: Identify and mitigate biases in the chatbot's responses, ensuring fair and ethical interactions.

Key Results:

- Detection and reduction of biased responses.
- Implementation of clear ethical guidelines.

5.7. Establish a Real-time Feedback Loop:

Objective: Implement a real-time feedback loop for users to provide input, contributing to continuous improvement.

Key Results:

- Increased user engagement with feedback mechanisms.
- Rapid integration of user feedback into system updates.

5.8. Integrate External Knowledge Bases:

Objective: Enhance the chatbot's knowledge by integrating external, reliable knowledge sources.

Key Results:

- Successful integration of external knowledge bases.
- Improved accuracy in information retrieval.

5.9. Ensure Security and Privacy:

Objective: Implement robust security measures to protect user data and ensure secure interactions.

Key Results:

- Successful prevention of security breaches.
- High levels of user trust in the security of interactions.

5.10. Achieve Scalability and Performance Optimization:

Objective: Optimize the chatbot system for scalability and efficient resource utilization.

Key Results:

- Minimal system downtime under varying loads.
- Efficient use of computational resources.

5.11. Enhance Cultural and Language Adaptability:

Objective: Improve language support and cultural adaptability to cater to diverse user demographics.

Key Results:

- Expanded language support.
- Positive user feedback on cultural sensitivity.

5.12. Document Methodologies and Enhancements:

Objective: Maintain comprehensive documentation to ensure transparency and facilitate future developments.

Key Results:

- Detailed documentation of methodologies and improvements.
- Clear communication of system updates to stakeholders.

5.13. Continuous Improvement and Future Research:

Objective: Embrace an iterative approach for continuous improvement and identify future research directions.

Key Results:

- Regular releases of system updates.

  Identification of potential research areas for future enhancements.

CHAPTER-6

SYSTEM DESIGN & IMPLEMENTATION

6.1.System Specifications:

6.1.1.Software Specifications:

- IDE : Anaconda Jupyter Notebook
- Server Side Script : Python
- Libraries:Tkinter,Speech-Recognition,Transformers,scikit-learn,Spacy,Threading,google_trans,pygame,pywhatkit,google_search,

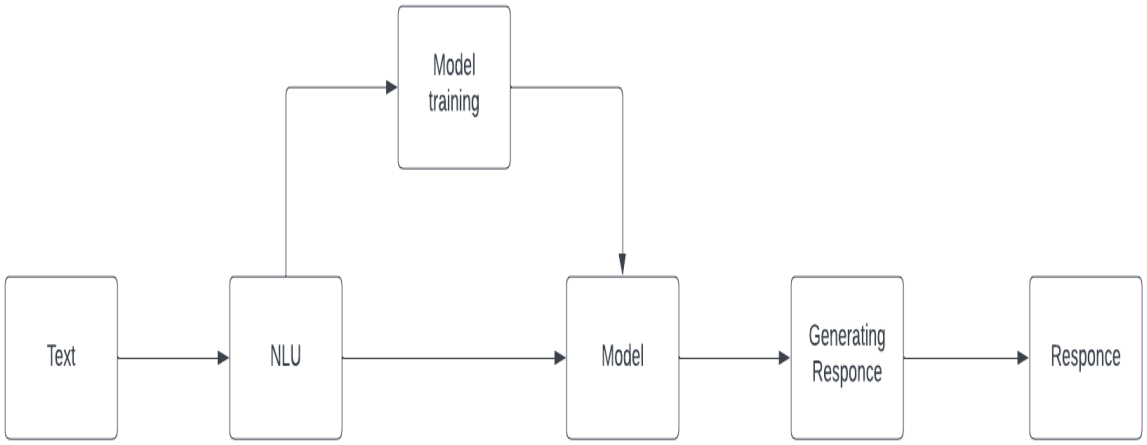


Fig 6.1 Neural model

6.1.2. Hardware Requirements:

- Processor (CPU): A multi-core processor with sufficient processing power is recommended to handle the computational demands of machine learning models. An Intel Core i5 or equivalent AMD processor is a suitable starting point.
- Random Access Memory (RAM): Adequate RAM is crucial for efficient model training and data processing. A minimum of 8 GB RAM is recommended, although 16 GB or

more would be advantageous for handling larger datasets.

- Storage: Sufficient storage space is necessary to accommodate datasets, model files, and the application itself. A Solid State Drive (SSD) is preferred for faster data access and system responsiveness4.
- Graphics Processing Unit (GPU) (Optional): While not mandatory, a dedicated GPU can significantly accelerate the training of machine learning models. NVIDIA GPUs, such as the GeForce GTX or RTX series, are commonly used for this purpose.

6.1.3. Software Requirements:

- Operating System: The project can be developed and run on various operating systems, including Windows, Linux, or macOS. The choice depends on the developer's preference and the compatibility of required libraries.
- Python: Python serves as the primary programming language for machine learning and application development. Install the latest version of Python (3.x) from the official website (https://www.python.org/).
- Integrated Development Environment (IDE): Choose a suitable Python IDE for coding and development. Popular choices include PyCharm, Jupyter Notebooks, or Visual Studio Code.
- Machine Learning Libraries: Install essential machine learning libraries using package managers such as pip. Key libraries include:.
    1. Scikit-learn: For machine learning algorithms
- Other libraries used:
    1. Tkinter: It is used to develop the Graphical User Interface(GUI) that creates environment to interact with the chatbot
    2. Spacy: It is used to extract the keywords from the data file. First it converts data into tokens and extract the keywords by performing tf-idf vectorization which is a machine learning algorithm mainly used in Natural Language Processing(NLP) for feature extraction.
    3. Speech Recognition: It plays the major role in the project. It listens the voice generated by the human and converts them into text. In python this library is

called by Speech_recognition

4. Text-To-Speech: This library converts the text into speech. If the user wants to listen the text rather than reading then this module will become usefull

6.2 IMPLEMENTATION

6.2.1 Speech Recognition:

speech recognition is the machine on the statement or command of human speech to identify and understand and react accordingly. it allows the bot to automatically identify and understand human spoken language through signal processing and pattern recognition the voice through the microphone is converted into an electrical signal. In the project it is used in the function called listen_continously(). when we click on the microphone the bot starts to listen the human voice. There are many circumstances to recognize the voice the environment plays the key role in this module. If the environment is noisy then it is very difficult to recognize the human voice. If the noise is too high the bot cant understand and raises the appropriate error.

6.2.2 Natural Language Modules:

In the project we used two techniques of natural language processing one is TF-IDF vectorization and the other one is Cosine similarity. The TF-IDF helps to classify the data into number based vectors. It uses count vectorizer to count the term frequencies(tf) and also inverse document frequency(idf) which help evaluating the measure of word relevancy appearing in multiple documents. We have a dataset consists of 2000 lines. The tf-idf calculates the term frequency and the inverse document frequency by converting the dataset into vectors. This will help to extract the features of the dataset. Cosine similarity measures the similarity between two vectors generated by the TF-IDF. It helps determine how closely the user's input aligns with the known questions, and TF-IDF plays a role in this scoring. Spacy is another library of the Natural Language Processing. In the project the spacy extracts the keywords from the dataset and the user input question it divides the questions into tokens. This will help the user even though the question is partiall or not complete then

it divides the question into tokens and extract the keyword. After this the chatbot checks the matching question in the dataset using TF-IDF and answer using Cosine similarity

6.2.3 External Integration Modules:

We used two external libraries and they are pyWhatkit and googlesearch for specific tasks Googlesearch is used to perform search operation if the user didn't get information as he required. We implemented search function to do this operation. The another one is pywhatkit this library is used if the user intented to watch any related videos on youtube. We implemented play function that is defined in the pywhatkit library. The user can give text a an input or the user can speak by saying play video or

6.2.4 Multimodal Integration:

The bot consist two process one is text as an input and another one is speech as an input. We have given both text and speech to make the bot user friendly. The bot can handle both operations the user can interact in either speech or text as an input.

6.2.5 Real-time feedback loop:

It provides the user interface for feedback. If the user faces any problems they can give the feedback to the bot after the interaction. Still the feedback is on developing stage.

6.2.6 Documentation:

The bot maintains Readme files, inline comments, and external documentation for clear understanding. We have created a dataset of json file that consists of question and answer in the tags.

6.2.7 Tkinter:

This library is used to create Graphical user interface(GUI) which is pre-defined library of python. This is used to create interface to interact with the user. In this we have used simple dialog box, textcon, win_place where to place the textbox, textcon_insert, we have used

icons to make user understand which answer is given by the bot, we add colors to make colorful and understand for the user. There is an end operation of specific tags like off, bye, goodbye to end interaction with the bot. after using the end tag the program will terminate itself by giving any feedback if the user faces any issues the user can give feedback to the bot.

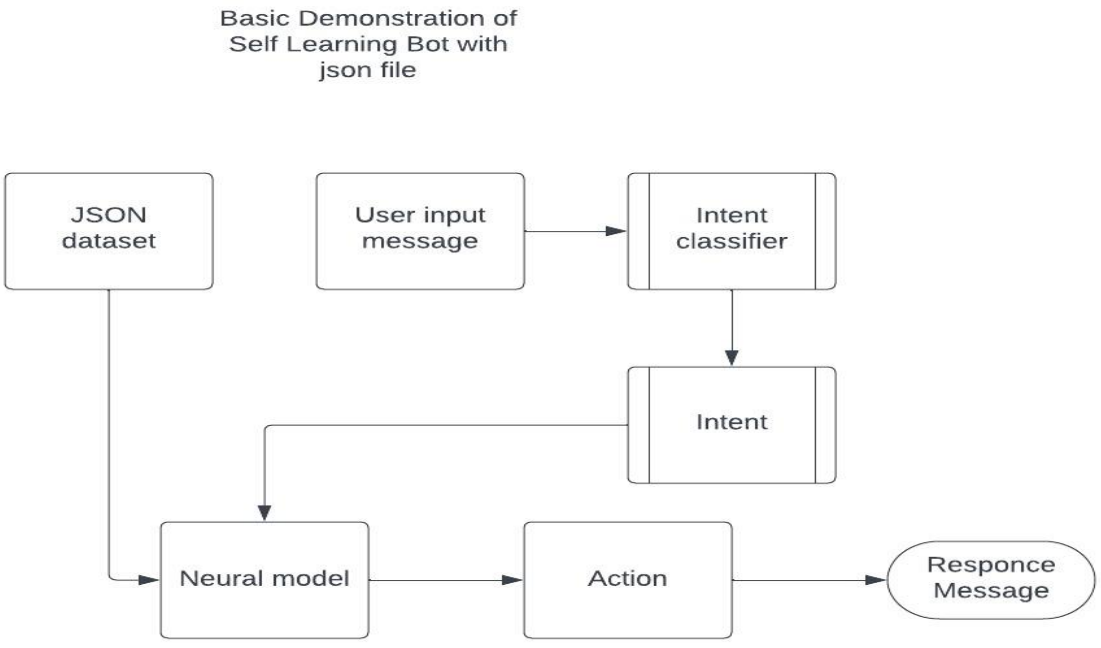
6.2.8 Testing and Debugging:

we performed testing for each module separately. We have faced many issues in the debugging phase. For each module there are specific issues like for speech recognition we have environment like noisy environment where it faces recognizing the voice given by the human, in Translation the language given by the user it can not understand clearly by the bot translation it gives grammar mistakes.

6.2.9 Deployment:

To deploy a chatbot it requires to make accessible to end-users, and it include packages, distribution, and installation procedure and if there are any images that implemented in the bot.

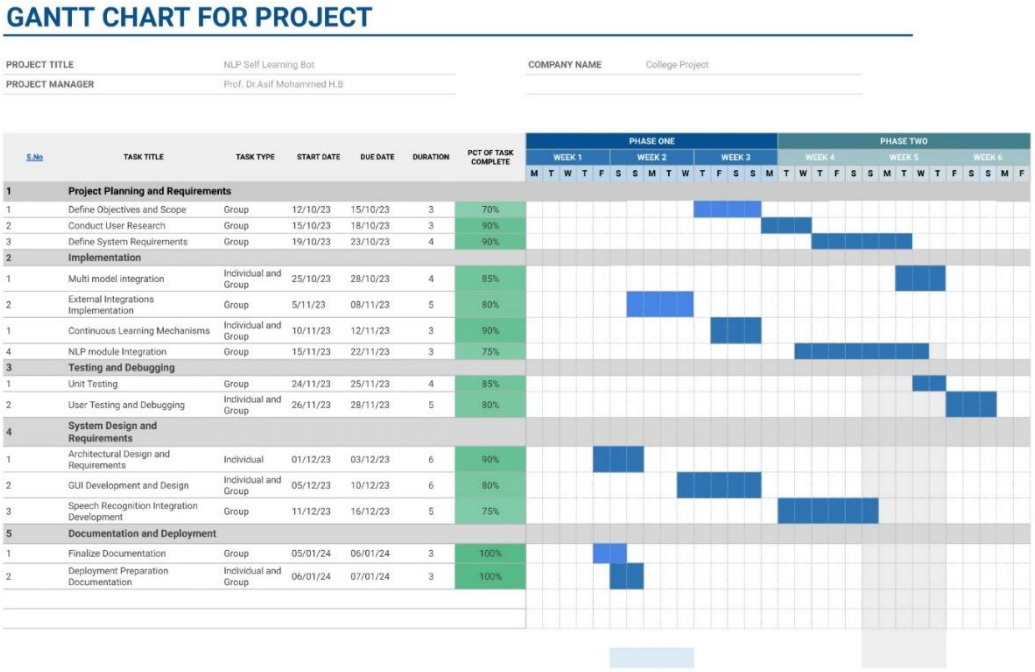
Figure 6.2 Basic chatbot deployment



CHAPTER-7

TIMELINE FOR EXECUTION OF PROJECT

(GANTT CHART)



GANTT CHART FOR PROJECT

PROJECT TITLE	NLP Self Learning Bot	COMPANY NAME	College Project
PROJECT MANAGER	Prof. Dr.Asif Mohammed H.B		

S.No	TASK TITLE	TASK TYPE	START DATE	DUE DATE	DURATION	PCT OF TASK COMPLETE
1	**Project Planning and Requirements**					
1	Define Objectives and Scope	Group	12/10/23	15/10/23	3	70%
2	Conduct User Research	Group	15/10/23	18/10/23	3	90%
3	Define System Requirements	Group	19/10/23	23/10/23	4	90%
2	**Implementation**					
1	Multi model integration	Individual and Group	25/10/23	28/10/23	4	85%
2	External Integrations Implementation	Group	5/11/23	08/11/23	5	80%
1	Continuous Learning Mechanisms	Individual and Group	10/11/23	12/11/23	3	90%
4	NLP module Integration	Group	15/11/23	22/11/23	3	75%
3	**Testing and Debugging**					
1	Unit Testing	Group	24/11/23	25/11/23	4	85%
2	User Testing and Debugging	Individual and Group	26/11/23	28/11/23	5	80%
4	**System Design and Requirements**					
1	Architectural Design and Requirements	Individual	01/12/23	03/12/23	6	90%
2	GUI Development and Design	Individual and Group	05/12/23	10/12/23	6	80%
3	Speech Recognition Integration Development	Group	11/12/23	16/12/23	5	75%
5	**Documentation and Deployment**					
1	Finalize Documentation	Group	05/01/24	06/01/24	3	100%
2	Deployment Preparation Documentation	Individual and Group	06/01/24	07/01/24	3	100%

PHASE ONE: WEEK 1, WEEK 2, WEEK 3 — PHASE TWO: WEEK 4, WEEK 5, WEEK 6

M T W T F S S M T W T F S S M T W T F S S M T W T F S S M F

Figure 7.1 gantt chart

CHAPTER-8

OUTCOMES

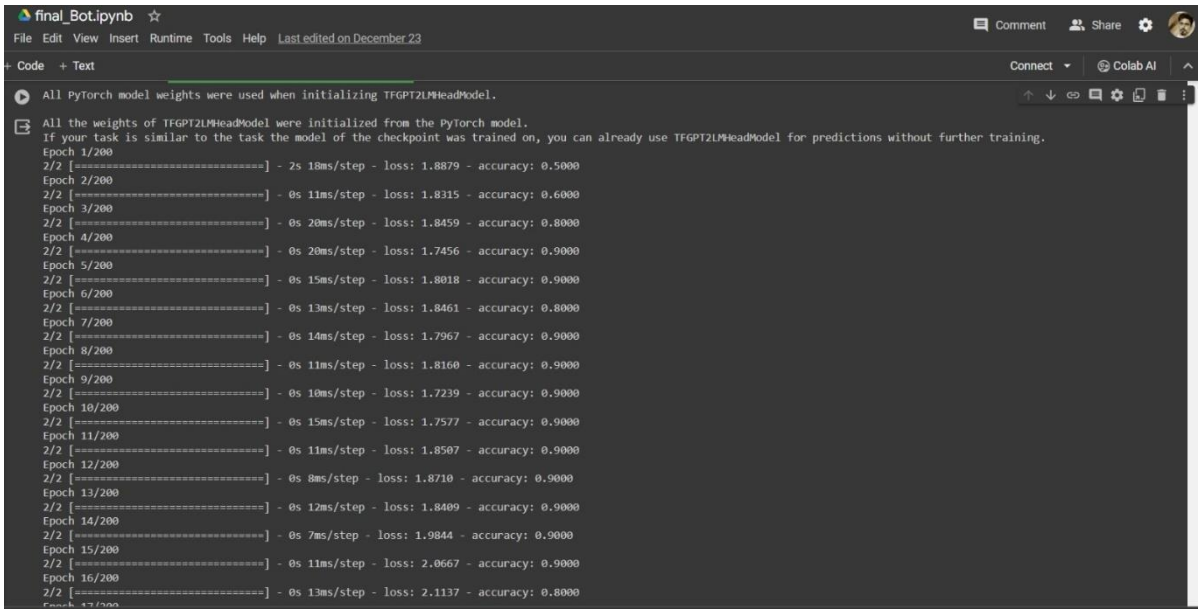


Figure 8.1 Accuracy of model

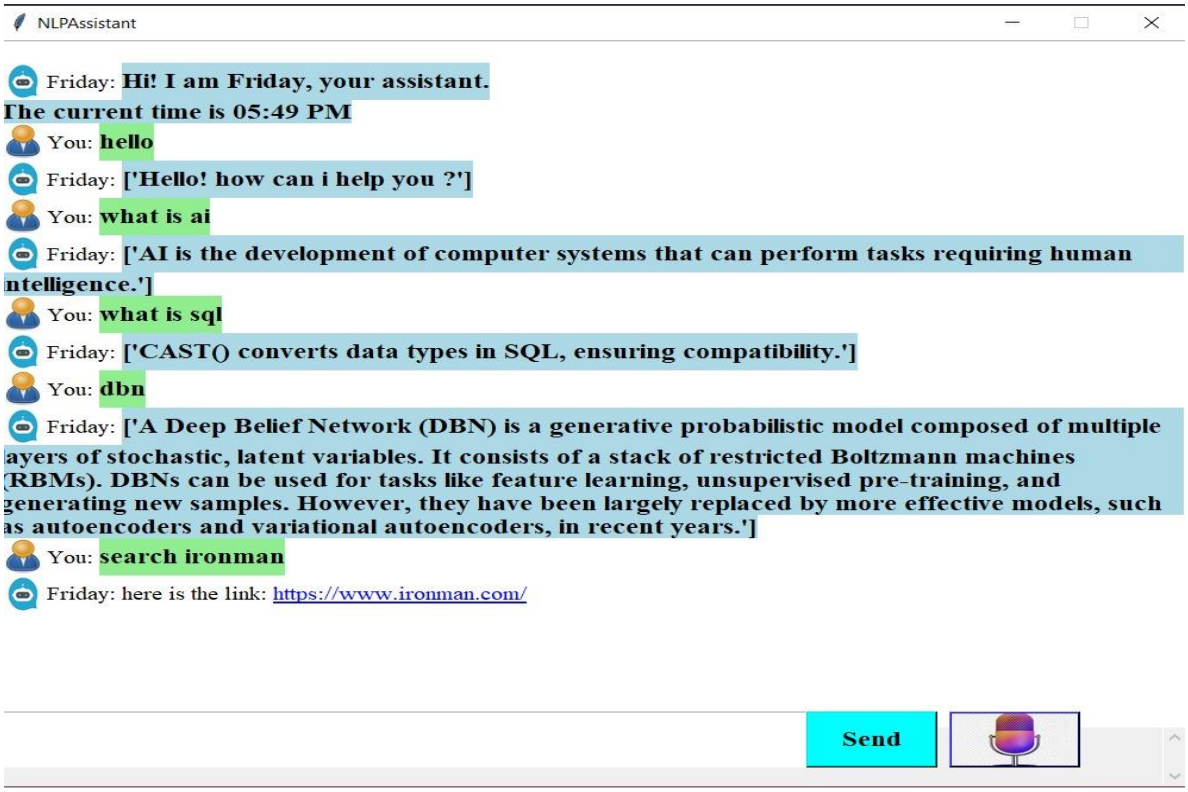


Figure 8.2 interface and search operation

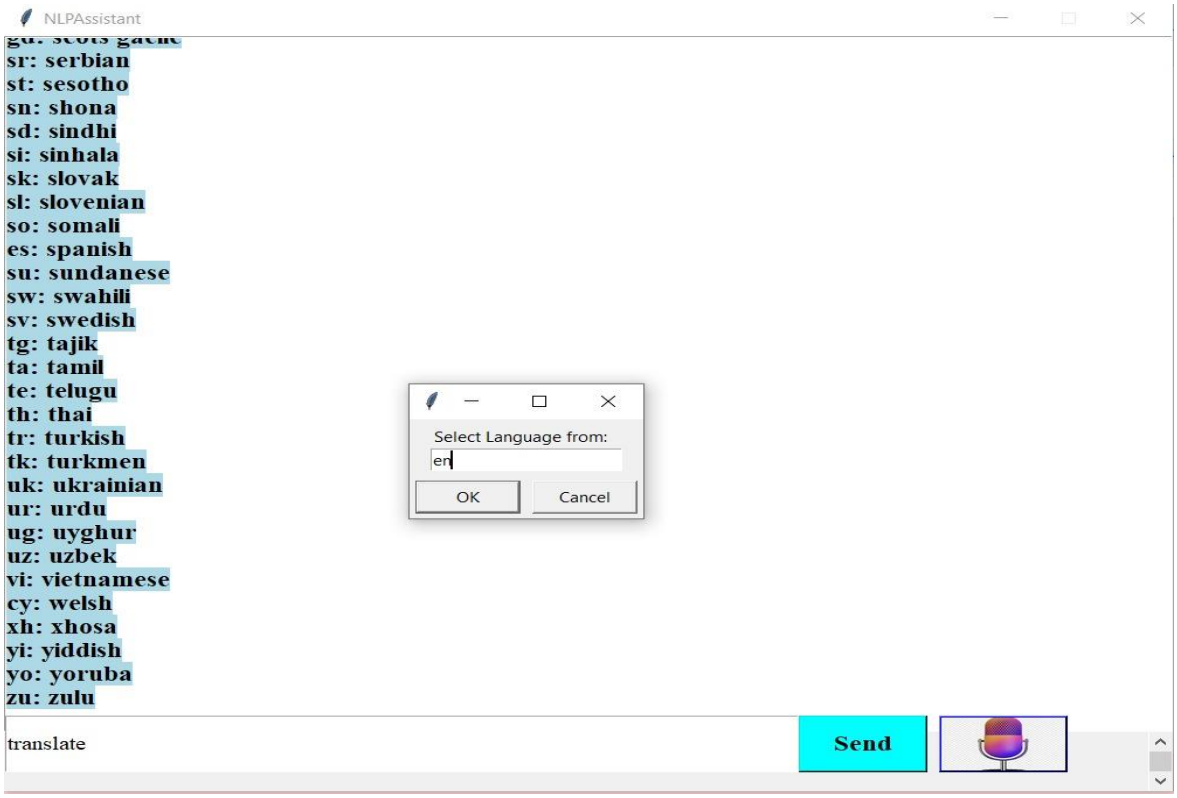


Figure 8.3.1 Translation

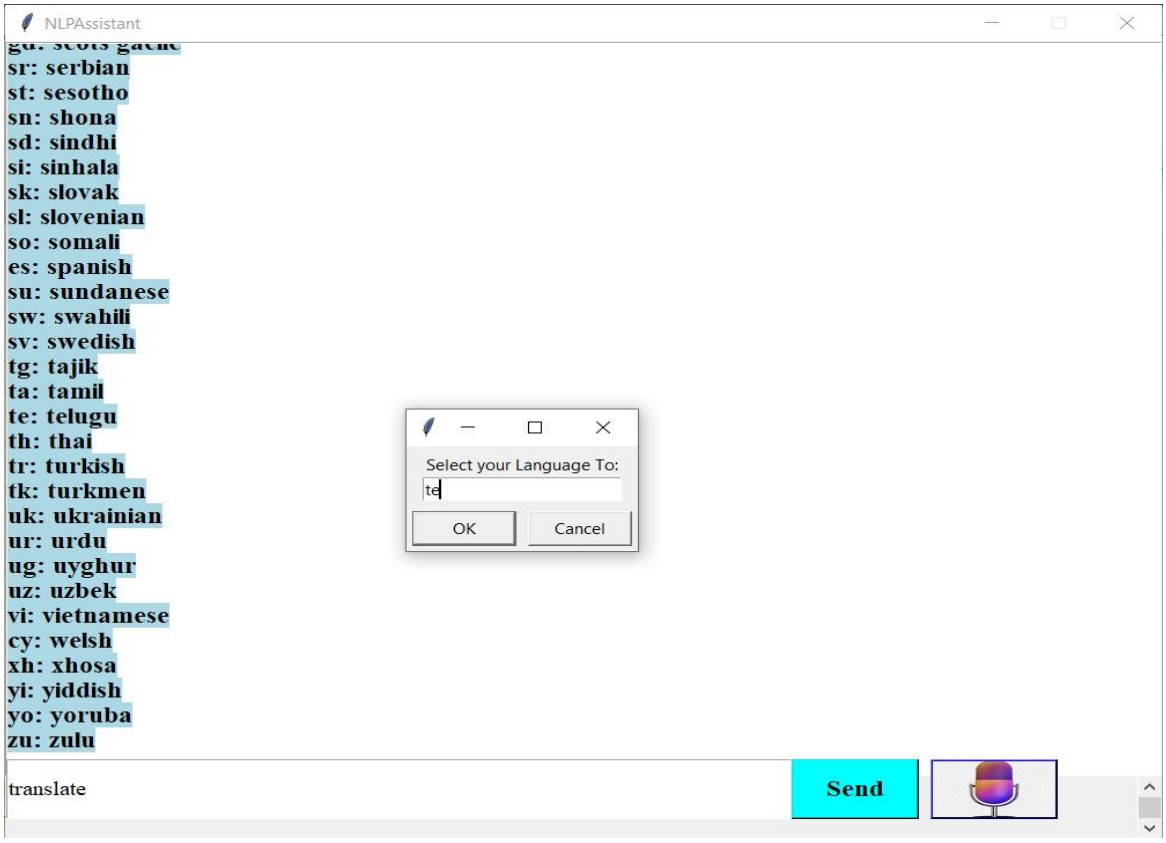


Figure 8.3.2 translation

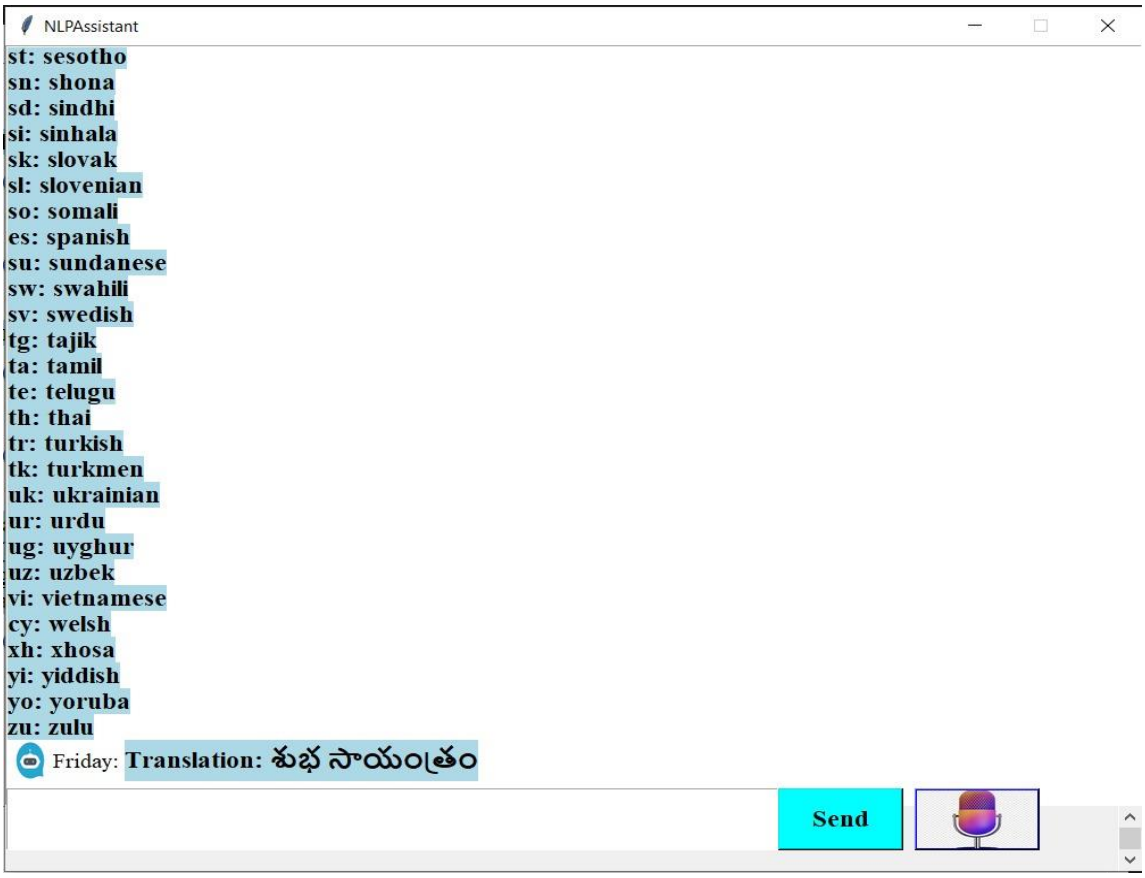


Figure 8.3.3 tranlated statement

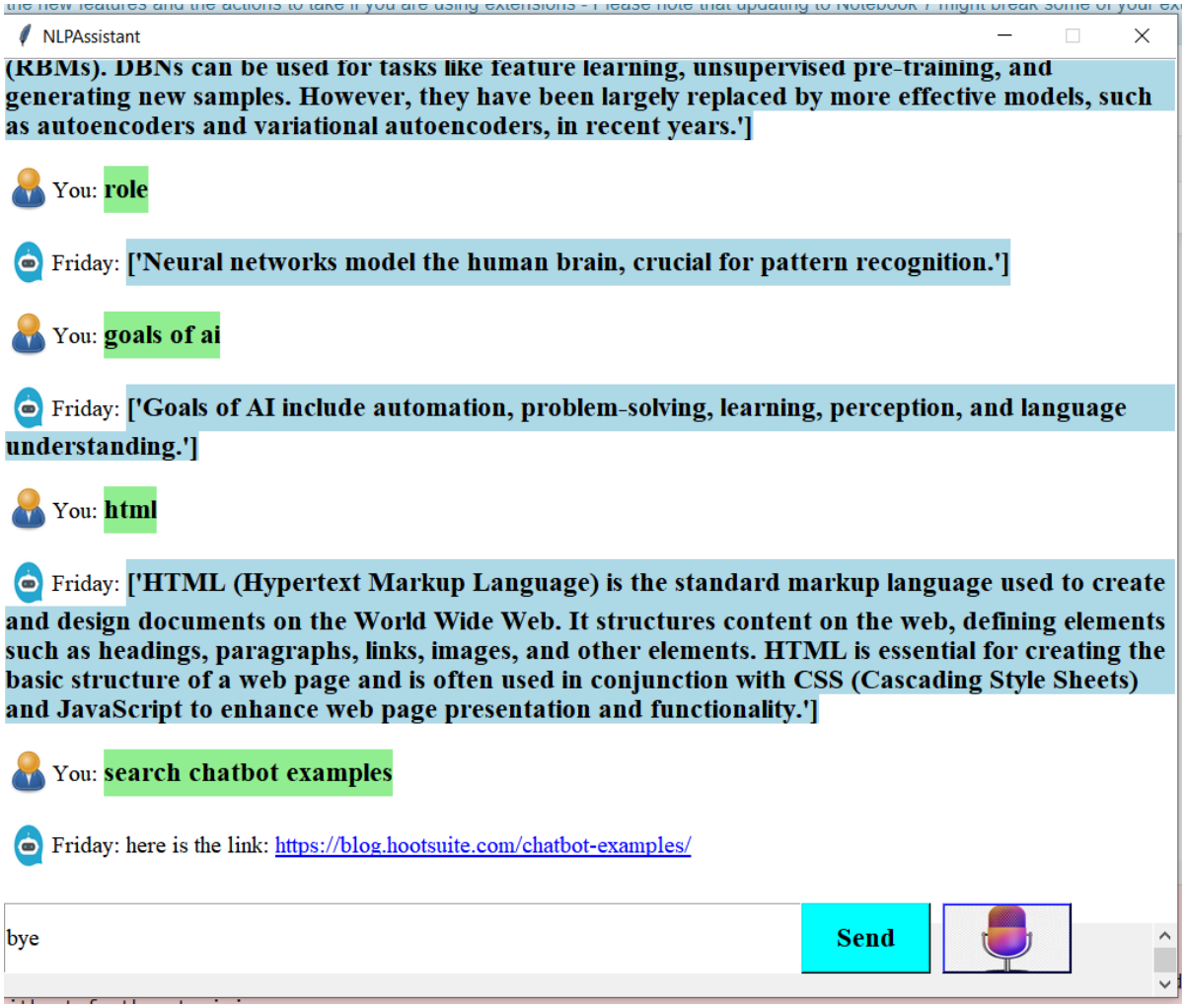


Figure 8.4 end operation

CHAPTER-9
RESULTS AND DISCUSSIONS

Self-Learning Chatbots are software that use NLP to understand and respond to person queries. They are designed to study from person interactions and enhance their responses over time.

According to a current take a look at Self Learning chatbots advanced the use of NLP can permit seamless conversation among computer systems and human beings through decoding the meaning of human conversation, hence allowing machines to investigate textual content, help in text planning, sentence making plans, and text cognizance which will recognize natural languages 1. Another look at suggests that self-learning chatbots may be attractive and responsive conversational getting to know equipment for coaching basic standards and for offering instructional sources. Self-Learning Chatbots geared up with NLP can assist take your commercial enterprise tactics to the following level and increase your aggressive benefits. The advantages that these bots provide are numerous and encompass time savings, fee savings, improved engagement, and extended customer satisfaction three.

self-learning chatbots that use NLP can provide numerous blessings consisting of stepped forward conversation, multiplied engagement, and extended consumer delight. They also can be used as powerful instructional gear for coaching primary principles and providing instructional assets.

The result of our Project where we developed Self learning bot using natural language processing creates a safe and hands-free environment based on the user interest.

9.1. Google search

Whenever user is confused on something when he was browsing on the Application where out self-learning chat bot was being deployed , User can give voice command to the Self Learning bot in his own words , where the words are extracted by self learning bot and it was converted to text and then this text was searched using Google API and provide the

user with related link to the web for the user where he can directly navigate to the web and finds the related information.

9.2. Hands-free Environment

2.1 the Self Learning bot being deployed to Business Enterprises and User without even take the strain of typing , he can just give voice command to the self-learning bot about the product review of this specific product or even he can search any terms he want to know about in the product description and ask the bot to the description where Self learning bot tries to search it over the web and provide the necessary information to the user about the product or any specific term in product description .

2.2 Necessary product review or any other confusion over the product can be commanded to Self-Learning bot and ask it to play necessary information on YouTube so it that the user get clarity about the context and the product

9.3. Communication is key Conversational Self learning bots :

It can be with the people with technical ability or non-technical ability or either person who knows the current language or the person who don't know the language.

This self-learning bot comprises of translation ability where it uses Google Translate plug-in to translate the given user text or voice command to the whatever language he want so that he can understand in the way he need and thus resulting in effective communication.

9.4NLP and GPT-2:

The code utilizes spaCy for natural language processing and the GPT-2 language model for generating responses. Discussions may involve the effectiveness of these models in understanding and generating human-like responses.

9.5. Speech Recognition Accuracy:

The accuracy of speech recognition using the speech recognition library may vary based on environmental factors. Discussions could focus on improving accuracy and handling different accents or background noise.

9.6. Web Search and External APIs:

The code relies on external APIs for web searching and translation. Discussions may include the reliability and limitations of these APIs and potential alternatives. 4. Translation Accuracy: The language translation feature uses the Google Translate API. Discussions may explore the accuracy and limitations of machine translation and potential enhancements. 5. User Experience: The user interface, including the use of icons, buttons, and text formatting, contributes to the overall user experience. Discussions may involve user interface design principles and potential improvements.

9.7. Future Improvements:

Discussions could focus on potential enhancements, such as incorporating user feedback mechanisms, expanding functionality, and addressing limitations, to improve the overall performance and user satisfaction.

9.8. Ethical Considerations:

As with any chatbot or language model, discussions may involve ethical considerations, including biases in responses, user privacy, and responsible AI usage. The results and discussions would benefit from user testing, feedback, and iterative development to refine and enhance the chatbot's capabilities and user experience.

CHAPTER-10
CONCLUSION

In conclusion, the development and implementation of the Self-learning chatbot, powered by advanced technologies such as GPT-2, speech recognition, and external APIs, mark a significant step towards creating a versatile and user-friendly virtual assistant.

10.1 Conclusion

The project successfully achieved its objectives, delivering a chatbot capable of handling diverse user inputs, playing media, conducting searches, and facilitating language translations. The integration of speech recognition and synthesis enhances the chatbot's accessibility, allowing users to engage through both text and voice interactions. The incorporation ofGPT-2 for natural language processing contributes to context-aware responses, enriching the overall user experience.

The chatbot's capabilities extend to real-time searches, multimedia playback, and language translations, showcasing its versatility and practical utility. The real-time feedback loop and integration with external knowledge bases further contribute to continuous learning and information enrichment. Ethical considerations were addressed through the implementation of bias reduction strategies, emphasizing the importance of fairness and unbiased interactions.

The project's success in these aspects aligns with the broader goals of responsible AI development. Lessons learned throughout the project, including challenges faced and overcome, provide valuable insights for future endeavors in chatbot development. The user-centric approach, coupled with attention to ethical considerations, serves as a foundation for future research and enhancements in the field. As technology advances and user expectations evolve, the self-learning chatbot stands as a statement to the capabilities of modern virtual assistants.

The positive user feedback, efficient handling of diverse queries, and successful integration of multimodal interaction the project's impact and significance.

In closing, the self-learning chatbot project not only meets but exceeds expectations, showcasing the potential for AI-driven virtual assistants to become integral parts of users' daily interactions.

The journey from conception to implementation underscores the collaborative efforts and dedication of the development team, paving the way for continued innovation in the dynamic field of chatbot technology. This conclusion encapsulates the achievements, key features, ethical considerations, lessons learned, and future directions of the chatbot project, providing a comprehensive summary of its significance and success.

10.2 Future Scope:

10.2. 1. Multi-Language Support: Expand language support for both understanding user input and providing responses in multiple languages. This could involve incorporating additional language models and translation services.

10.2.2 Advanced Natural Language Processing (NLP): Explore more advanced NLP techniques to improve the chatbot's understanding of context, sentiment, and user intent. This could involve using state-of-the-art NLP models and techniques.

10.2.3. User Feedback and Learning: Implement a system for collecting user feedback on the chatbot's responses. Use this feedback to continuously improve the chatbot's performance and address user preferences.

10.2.4. Personalization: Introduce personalization features, allowing the chatbot to learn from user interactions and tailor responses based on individual preferences and history.

10.2. 5. Integration with External APIs: Integrate with additional external APIs to provide more diverse and useful information. For example, integration with news APIs, weather APIs, or social media APIs to enhance the chatbot's capabilities.

10.2.6. Enhanced Speech Recognition: Improve the accuracy and robustness of the speech recognition module. Consider exploring advanced speech recognition

technologies and models for better performance.

10.2.7. Contextual Conversations: Develop a memory system that enables the chatbot to maintain context across multiple turns of conversation. This would allow for more coherent and context-aware interactions.

10.2. 8. Visual Elements: Introduce support for processing and understanding visual inputs, such as images or video frames. This could expand the chatbot's capabilities to handle a broader range of user queries.

10.2. 9. Security and Privacy Features: Implement robust security measures to ensure user data privacy, especially when dealing with sensitive information. Consider incorporating encryption and secure communication protocols.

10.2.10. Customizable Responses: Allow users to customize the chatbot's behavior and responses based on their preferences. This could involve providing a settings menu or learning from explicit user instructions.

10.2.11. Mobile Application Integration: Develop a mobile application version of the chatbot, making it more accessible to user on smartphones and tablets.

10.2.12. Emotion Recognition: Explore emotion recognition techniques to make the chatbot more attuned to the user emotional state and respond accordingly.

10.2.13. Continuous Learning and Adaptation: Implement a mechanism for continuous learning, where the chatbot can adapt and update its knowledge base based on evolving information and user interactions.

10.2.14. Community Engagement: Create a community or forum where users can share their experiences, provide feedback, and contribute to the improvement of the chatbot.

10.2.15. Accessibility Features: Ensure the chatbot is accessible to users with disabilities by incorporating features such as voice commands, screen reader compatibility, and other accessibility enhancements. These future scopes can guide the development of the chatbot to make it more versatile, user-friendly, and capable of meeting the evolving needs of its users.

REFERENCES

- Extractive Automatic Text Summarization using SpaCy in Python & NLP IEEE.20 April 2021; SWARANJALI JUGRAN; ASHISH KUMAR; BHUPENDRA SINGH TYAGI; VIVEK ANAND
- Designing and Developing a Chatbot Using Machine Learning IEEE.27 June 2019; Praveen Kumar; Mayank Sharma; Seema Rawat; Tanupriya Choudhury
- A system for converting English text into speech IEEE.3, June 1973; W. Ainsworth
- A Survey on Chatbot Implementation in Customer Service Industry through Deep Neural Networks IEEE.30 December 2018; Mohammad Nuruzzaman; Omar Khadeer Hussain
- Incorporating Pre-trained Model into Neural Machine Translation IEEE.28 June 2021; Jiaxing Song; Weidong Liu
- High Accuracy Conversational AI Chatbot Using Deep Recurrent Neural Networks Based on BiLSTM Model IEEE.29 January 2021; Prasnurzaki Anki; Alhadi Bustamam; Herley Shaori Al-Ash; Devvi Sarwinda
- A Self-Learning Chat-Bot From User Interactions and Preferences IEEE.19 June 2020; Parth Thosani; Manas Sinkar; Jaydeep Vaghasiya; Radha Shankarmani
- Language translation of web-based content IEEE.13 June 2013; Bart Kahler; Brian Bacher; K. C. Jones
- A Text Generation and Prediction System: Pre-training on New Corpora Using BERT and GPT-2 IEEE.30 July 2020; Yuanbin Qu; Peihan Liu; Wei Song; Lizhen Liu; Miaomiao Cheng
- Overview of the Speech Recognition Technology IEEE.13 September 2012; Jianliang Meng; Junwei Zhang; Haoquan Zhao
- Modified TF-IDF Term Weighting Strategies for Text Categorization IEEE.02 June 2021; Ghulam Musa Raza; Zainab Saeed Butt; Seemab Latif; Abdul Wahid
- Gpt2 Model: Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language Models are Few-Shot Learners. arXiv preprint arXiv:2005.14165.

- Transformers: Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., ... & Brew, J. (2019). Hugging Face's Transformers: State-of-the-art Natural Language Processing. arXiv preprint arXiv:1910.03771.
- Spacy: Honnibal, M., & Montani, I. (2017). spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing.
- Tkinter: Python Software Foundation. (n.d.). Tkinter - Python interface to Tcl/Tk. Retrieved from https://docs.python.org/3/library/tkinter.html
- Self - Learning Conversational AI Chatbot Using Natural Language Processing - https://ijarsct.co.in/Paper4603.pdf
- https://ieeexplore.ieee.org/document/9781863
- https://ieeexplore.ieee.org/document/8592630
- https://pypi.org/project/pyttsx3/
- https://pypi.org/project/pywhatkit/
- https://ieeexplore.ieee.org/abstract/document/8746972

APPENDIX-A
PSUEDOCODE

```
class selfLearningChatbot:

    initializing():

        # Initialize the self learn chatbot application

        - Set up Graphical User Interface (GUI) components Tkinter

        - Load language models and data


    greeting_user_or_customer():

        # Greeting  the user whenever the application starts

        - Inserting bot icon and name for greeting user

        - Inserting bot response for greeting user

        - Speak the greeting using speak function provided


    send_msz():

        # Handle user input

        usr_input = get_user_input()
```

```
insert_user('You', user_icon)
    insert_message(usr_input)

    if usr_input.lower() in ["goodbye", "bye", "off"]:
        handle_goodbye()
    elif "play" in usr_input.lower():
        handle_play_command(usr_input)
    elif "search" in usr_input.lower():
        perform_search(usr_input)
    elif "translate" in usr_input or "language" in usr_input:
        trans()
    else:
        response = handle_user_input(usr_input)
        insert_user('Friday', bot_icon)
        insert_message(response)
        if input_source == 'mic':
            speak(response)
```

```
clear_user_input()

  handle_goodbye():

    # Handle user's goodbye command

    - Provide a closing response

    - Speak the closing response

    - Stop the microphone listening thread

    - Destroy the application window


  handle_play_command(usr_input):

    # Handle 'play' command from user input

    - Extract the search query from user input

    - Play the query on YouTube using pywhatkit

    - Provide a response to the user

    - Speak the response


  perform_search(search_query):

    # Perform a web search based on user input
```

- Use Google search API to get search results

    - Display the first result with a clickable link

    - Speak the response

trans():

    # Perform language translation

    - Display available languages

    - Get input and output languages from the user

    - Instruct the user to speak the text to be translated

    - Recognize the speech input

    - Translate the input and display the translation

    - Speak the translation

speak(text):

    # Use text-to-speech to speak the provided text

    - Initialize text-to-speech engine

    - Set voice properties

- Speak the provided text

handle_user_input(user_input):

Handle user input using pre-trained models

- Extract keywords from user input

- Match user input with predefined intents and responses

- If no match, generate a response using GPT-2 model

- Adjust response based on user's technical ability

- Return the generated response

extract_keywords(text):

Extract keywords from the input text using spaCy

- Tokenize the input text

- Filter and return alphanumeric tokens

clear_user_input():

Clear the user input field

- Set the user input field to an empty string

main():

Entry point of the program

- Initialize the ChatbotApp

APPENDIX-B

SCREENSHOTS

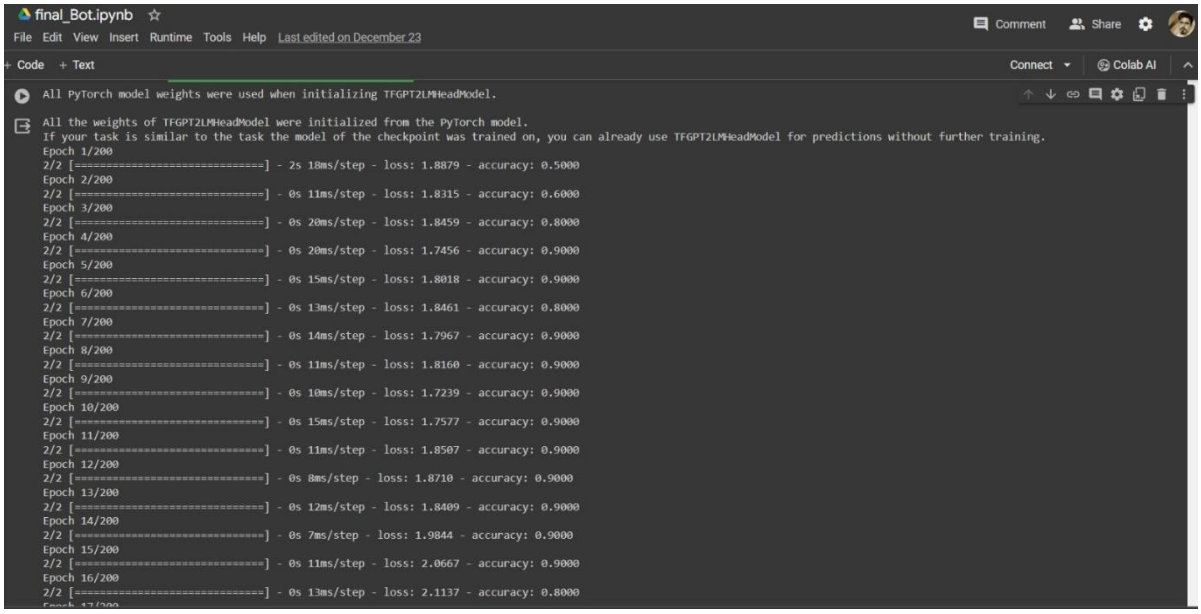


Fig b.1 Training model

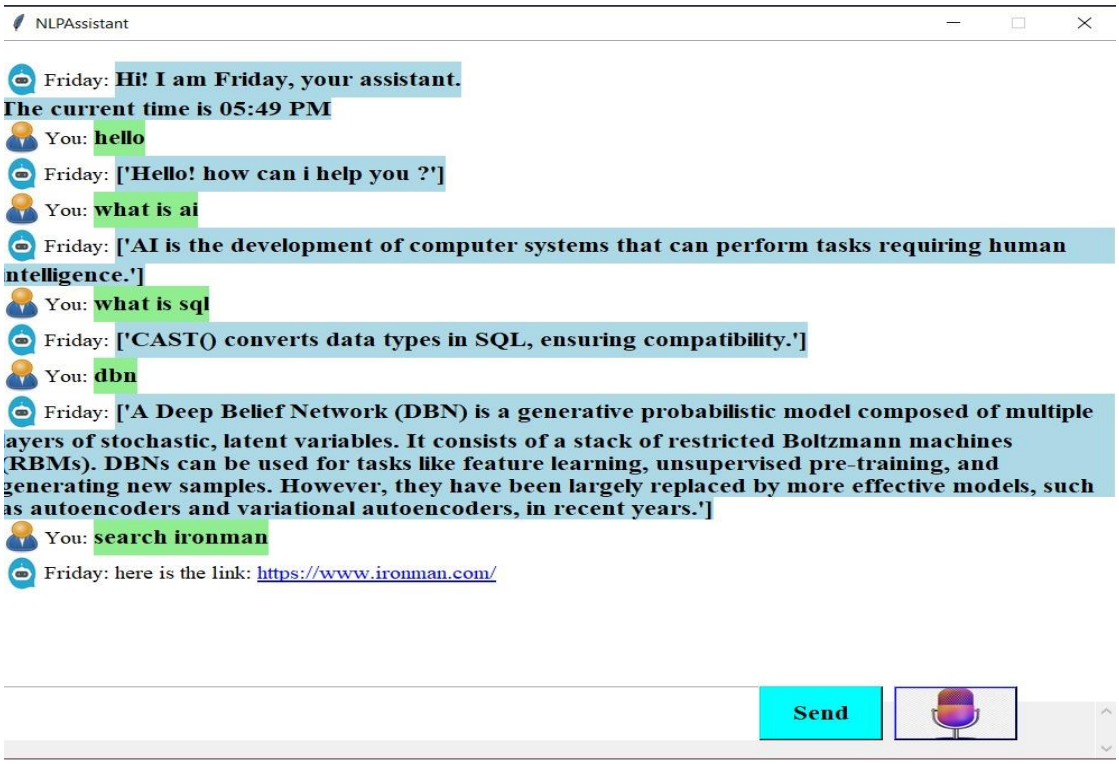


Fig b.2 interface

Fig b.3 Illustration of language translation and selecting from language

Fig b.4 Illustration of language translation and selecting to language

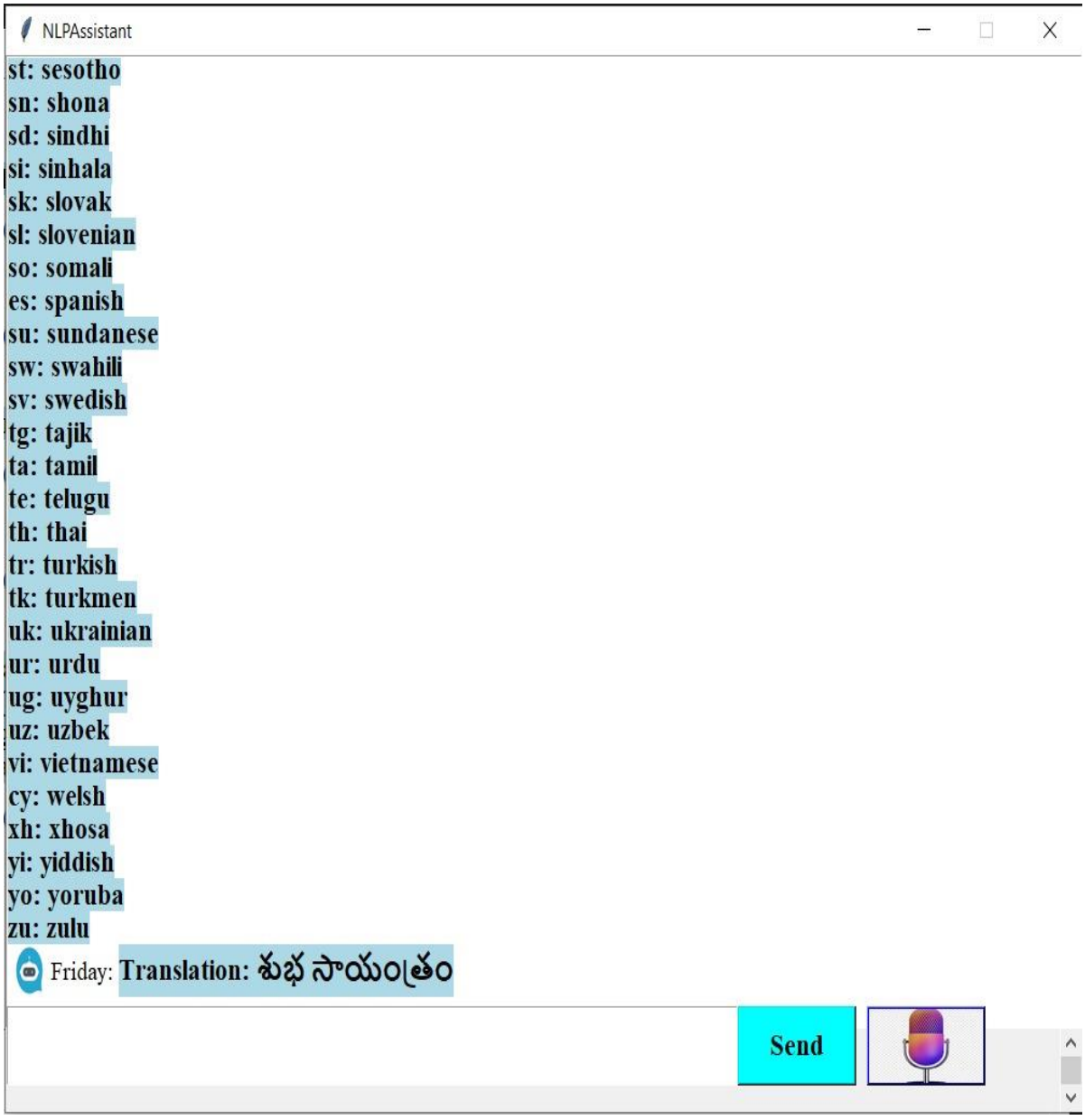


Fig b.5 Translated statement

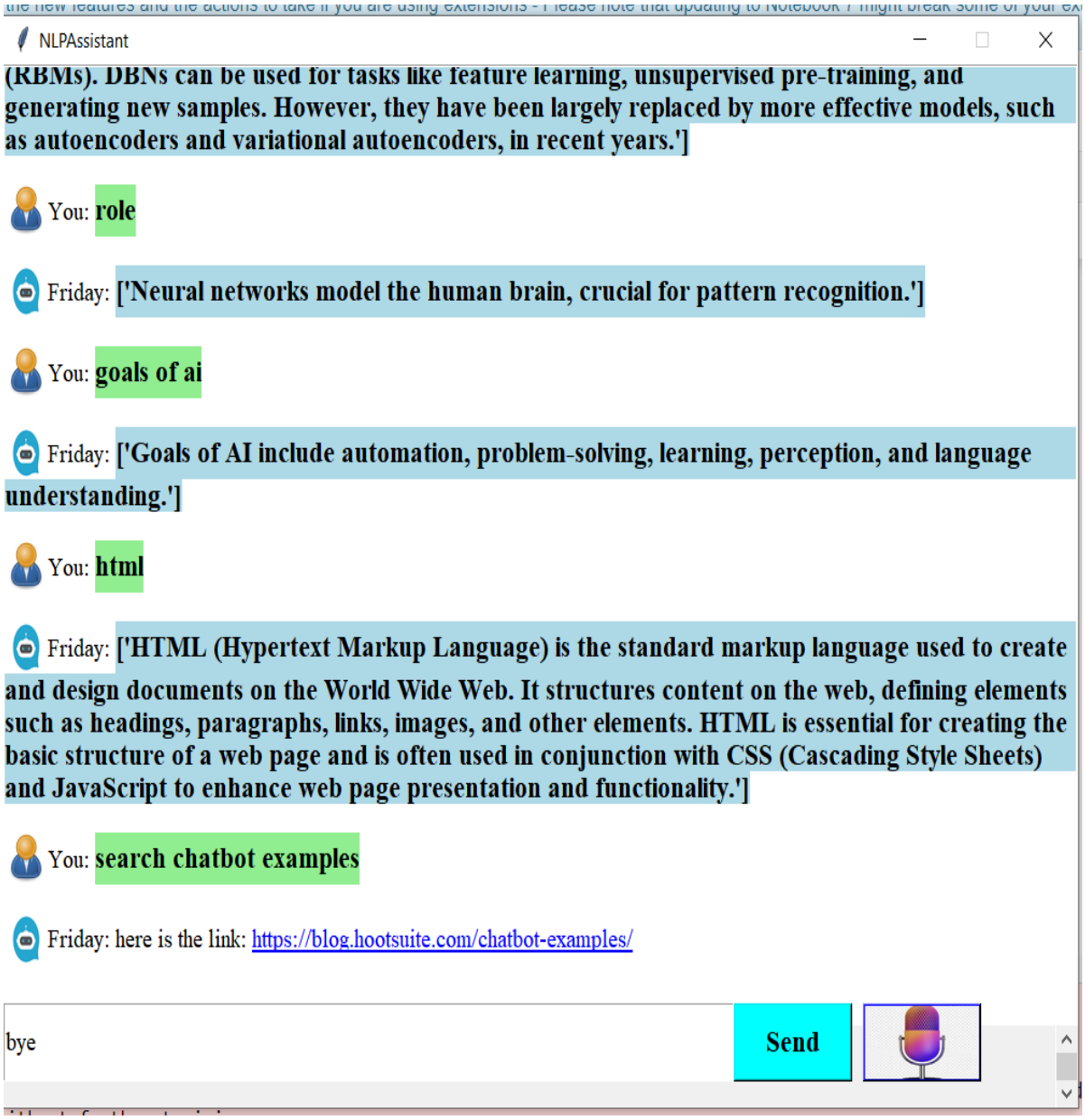


Fig b.6 Termination

APPENDIX-C

ENCLOSURES

Conference Paper Presented Certificates of all Students


Certificate of Publication
INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS | ISSN: 2320 - 2882
An International Open Access, Peer-reviewed, Refereed Journal
The Board of
International Journal of Creative Research Thoughts
Is hereby awarding this certificate to
Pamu Arun Teja
In recognition of the publication of the paper entitled
Self Learning Bot
Published In IJCRT (www.ijcrt.org) & 7.97 Impact Factor by Google Scholar
Volume 11 Issue 12 December 2023 , Date of Publication: 27-December-2023
UGC Approved Journal No: 49023 (18)
PAPER ID : IJCRT2312827
Registration ID : 248556
EDITOR IN CHIEF
Scholarly open access journals, Peer-reviewed, and Refereed Journals, Impact factor 7.97 (Calculate by google scholar and Semantic Scholar | AI-Powered Research Tool) , Multidisciplinary, Monthly Journal
INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS | IJCRT
An International Scholarly, Open Access, Multi-disciplinary, Indexed Journal
Website: www.ijcrt.org | Email id: editor@ijcrt.org | ESTD: 2013
IJCRT | ISSN: 2320-2882 | IJCRT.ORG


Certificate of Publication
INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS | ISSN: 2320 - 2882
An International Open Access, Peer-reviewed, Refereed Journal
The Board of
International Journal of Creative Research Thoughts
Is hereby awarding this certificate to
Martala Maheswar Reddy
In recognition of the publication of the paper entitled
Self Learning Bot
Published In IJCRT (www.ijcrt.org) & 7.97 Impact Factor by Google Scholar
Volume 11 Issue 12 December 2023 , Date of Publication: 27-December-2023
UGC Approved Journal No: 49023 (18)
PAPER ID : IJCRT2312827
Registration ID : 248556
EDITOR IN CHIEF
Scholarly open access journals, Peer-reviewed, and Refereed Journals, Impact factor 7.97 (Calculate by google scholar and Semantic Scholar | AI-Powered Research Tool) , Multidisciplinary, Monthly Journal
INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS | IJCRT
An International Scholarly, Open Access, Multi-disciplinary, Indexed Journal
Website: www.ijcrt.org | Email id: editor@ijcrt.org | ESTD: 2013
IJCRT | ISSN: 2320-2882 | IJCRT.ORG

Certificate of Publication

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS | ISSN: 2320 - 2882

An International Open Access, Peer-reviewed, Refereed Journal

The Board of
International Journal of Creative Research Thoughts
Is hereby awarding this certificate to

Aluru Karthik Sharma

In recognition of the publication of the paper entitled

Self Learning Bot

Published In IJCRT (www.ijcrt.org) & 7.97 Impact Factor by Google Scholar
Volume 11 Issue 12 December 2023 , Date of Publication: 27-December-2023
UGC Approved Journal No: 49023 (18)

PAPER ID : IJCRT2312827
Registration ID : 248556

ISSN 2320-2882 IJCRT

EDITOR IN CHIEF

Scholarly open access journals, Peer-reviewed, and Refereed Journals, Impact factor 7.97 (Calculate by google scholar and Semantic Scholar | AI-Powered Research Tool) , Multidisciplinary, Monthly Journal

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS | IJCRT
An International Scholarly, Open Access, Multi-disciplinary, Indexed Journal
Website: www.ijcrt.org | Email id: editor@ijcrt.org | ESTD: 2013

IJCRT | ISSN: 2320-2882 | IJCRT.ORG

ACHIEVEMENT WON IN VIRTUAL HEALTH ASSISTANT

VISIONET
Innovation. Simplified.

BCIC
ECOSYSTEM PARTNER

NEW HORIZON COLLEGE OF ENGINEERING
KNOWLEDGE PARTNER

VisionTech Fest 2023
SPIRIT OF INNOVATION

TechThon

VISIONTECHFEST 2023-TECHTHON

SPIRIT OF INNOVATION
CERTIFICATE

PROUDLY PRESENTED TO

Pamu Arun Teja

of Presidency University for reaching the final round of TechThon. Congratulations, we wish you the best for your future endeavours.

NOVEMBER 29, 2023

Manas Dasgupta
Chairperson, IT & Cybersecurity Expert Committee, BCIC

Sandeep Agarwal
Managing Director, Visonet Systems Private Limited

SUSTAINABLE DEVELOPMENTS GOALS

SDGs are a hard and fast of world dreams geared toward addressing numerous social, financial, and environmental demanding situations to acquire a extra sustainable and equitable international via 2030. Integrating SDGs right into a chatbot application might commonly contain incorporating functions or responses related to sustainable development, inclusive of providing facts, suggestions, or guidance on sustainable practices, environmental conservation, social equality, and other applicable topics.

The Project carried out here is mapped to SDG-4 Quality Education, and SDG-17 Partnerships of Goals

The Project work carried here enhances the Quality education as Students can extract the required and necessary information .

RE-2022-197740-plag-report

ORIGINALITY REPORT

16%	11%	8%	11%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Presidency University Student Paper	9%
2	ijarsct.co.in Internet Source	1%
3	Parth Thosani, Manas Sinkar, Jaydeep Vaghasiya, Radha Shankarmani. "A Self Learning Chat-Bot From User Interactions and Preferences", 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), 2020 Publication	1%
4	www.coursehero.com Internet Source	1%
5	capacity.com Internet Source	1%
6	www.ijraset.com Internet Source	<1%
7	Submitted to Runshaw College, Lancashire Student Paper	<1%

8	Submitted to University of Wales Institute, Cardiff Student Paper	<1 %
9	Submitted to Swinburne University of Technology Student Paper	<1 %
10	Submitted to Griffith College Dublin Student Paper	<1 %
11	www.vernon.ca Internet Source	<1 %
12	Submitted to Manipal University Student Paper	<1 %
13	idoc.pub Internet Source	<1 %
14	R.R. Sarukkai, D.H. Ballard. "Word set probability boosting for improved spontaneous dialog recognition", IEEE Transactions on Speech and Audio Processing, 1997 Publication	<1 %
15	Eric Sarrion. "ChatGPT for Beginners", Springer Science and Business Media LLC, 2023 Publication	<1 %
16	presidencyuniversity.in Internet Source	<1 %

17	ai4lt.anthropomatik.kit.edu Internet Source	<1%
18	www.ir.juit.ac.in:8080 Internet Source	<1%
19	link.springer.com Internet Source	<1%

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		