GOV BOT-A HEALTH SCHEME ASSIST

A PROJECT REPORT

Submitted by

JANANI V 210701087

JAYADARSHINI V 210701088

KEERTHANA H 210701117

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

ANNA UNIVERSITY, CHENNAI

MAY 2024

RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI

BONAFIDE CERTIFICATE

Certified that this Thesis titled **"GOV BOT- A HEALTH CARE ASSISTANT**" is the bonafide work of "**JANANI V (2116210701087), JAYADARSHINI V (2116210701088), KEERTHANA H (2116210701117)"** who carried out the work under my supervision. Certified furtherthat to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred onan earlier occasion on this or any other candidate.

SIGNATURE

Dr. K.Ananth M.E., Ph.D.,

PROJECT COORDINATOR

Professor

Department of Computer Science and Engineering

Rajalakshmi Engineering College

Chennai - 602 105

Submitted to Project Viva-Voce Examination held on_____

Internal Examiner **External Examine**r

ABSTRACT

This project evolves around the development of a chatbot, that helps individuals to explore and understand government health policies relevant to their circumstances. This aims to helps to imply various modules, by which the chatbot analyses the user inputs and provides personalized recommendations, enhancing accessibility and awareness of the supportive measures given by the government. The complexities of government health programs can leave people feel unable to access the policies that are given by them to utilise the benefits provided by them. The development of chatbot acts as a user-friendly and informative guide, helps in assisting the individuals by identifying potential programs based on user-provided details what kind of policies they are looking for like the disabilities that they have or the disease that they have directing users to official government and reputable health organization websites for detailed information and application processes. This chatbot helps the user to find the schemes that they are particularly looking for .For example if the user needs to find about the pregnancy schemes that are available then they can give the input which includes the keyword “PREGNANCY” then the chatbot will recognize the keyword and will provide the necessary schemes related to it .This chatbot is created using the GUI which facilitates interaction between the user and the system. This project strives to bridge the gap between policy and people.

.

ACKNOWLEDGMENT

First, we thank the almighty god for the successful completion of the project. Our sincere thanks to our chairman **Mr. S. Meganathan B.E., F.I.E.,** for his sincere endeavor in educating us in his premier institution. We would like to express our deep gratitude to our beloved Chairperson **Dr. Thangam Meganathan Ph.D.,** for her enthusiastic motivation which inspired us a lot in completing this project and Vice Chairman **Mr. Abhay Shankar Meganathan B.E., M.S.,** for providing us with the requisite infrastructure.

We also express our sincere gratitude to our college Principal, **Dr. S. N. Murugesan M.E., PhD.,** and **Dr. P. KUMAR M.E., PhD, Director computing and information science , and Head Of Department of Computer Science and Engineering** and our project coordinator **Dr. K.Ananth M.E.,Ph.D.,** for her encouragement and guiding us throughout the project towards successful completion of this project and to our parents, friends, all faculty members and supporting staffs for their direct and indirect involvement in successful completion of the project for their encouragement and support.

JANANI V

JAYADARSHINI V

KEERTHANA H

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In the developing society, the efficiency and accessibility of governmental policies play a critical role in improving the well-being and prosperity of the individuals and communities. Individuals seeking vital resources to support their health and well-being can find themselves overwhelmed by complex eligibility requirements and a lack of readily available information. The sheer volume and the complexities of the policies can make people feel uncomfortable and it may also act as a barrier to the individuals seeking assistance.

The development of artificial intelligence (AI) technologies offers promising techniques for addressing this need. This project proposes the development of a user-friendly chatbot, designed to bridge this knowledge gap and empower individuals to navigate the government health policies. The chatbot aims to act as a informative guide, simplifying complex information and assisting users in identifying and understanding programs that could be relevant to their specific circumstances, providing information, the Government Scheme Bot also facilitates the application process for eligible individuals.

Chatbots, in particular, have come up as versatile tools capable of helping users in natural language interactions to deliver information and assistance. By increasing the power of machine learning algorithms, chatbots can analyze user inputs, understand users preferences and circumstances, and provide personalized recommendations to them. Through interactive interfaces, users can complete their applications easily, without much difficulty in the understanding the policies and the criteria.

Furthermore, the chatbot also employs robust security methods to safeguard user data and ensure privacy regulations, thereby ensuring trust and confidence among users in their interactions with the chatbot. The current challenges associated with accessing government health programs are multifaceted. Each program has its own eligibility criteria and application processes, making it difficult for individuals to determine which programs they qualify for and how to participate. By demystifying policy complexities and empowering users with readily available information, Chatbot aims to improve access to valuable healthcare resources and foster a more informed and proactive citizenry when it comes to managing health.

1.1 PROBLEM STATEMENT

The complex nature of government health policies creates a knowledge gap for individuals seeking vital healthcare resources. This results in underutilization of programs due to confusion and lack of guidance, as well as a fear of bureaucratic application processes. This project aims to bridge this gap by developing a user-friendly solution that empowers individuals to navigate the healthcare landscape with confidence.

1.2 SCOPE OF THE WORK

This project focuses on developing a user-friendly chatbot, to help individuals in navigating government health policies. Chatbot will help to gather user information like age, income, and health conditions and by analyzing this data, the chatbot will identify potentially relevant programs such as Medicare or disability benefits so that the user can avail the relevant policy without any hesitation. Chatbot's scope is limited to providing information but the application process has to be handled by the user in the official website by them. It includes GUI which is user-friendly.

1.4 AIM AND OBJECTIVES OF THE PROJECT

This project aims to empower individuals to navigate government health policies through a user-friendly chatbot. Chatbot seeks to increase healthcare resource access, bridge the gap based on the programs, and empowering the individuals to take charge of their health by providing clear information and a user-friendly experience.

This objective ensures that Chatbot is user-friendly and accessible to a wide range of people. It focuses on helping people to develop a clear idea of the policy and to gain perfect insight of what they can gain out of it. By implementing the machine learning algorithms, the chatbot access the information of the user and provide them with the information regarding the policies of the government that they can avail without any hesitation.

1.5 RESOURCES

This project has been developed through widespread secondary research of accredited manuscripts, standard papers journals and conference reviews. The following prospectus details a list of resources that will play a primary role in the successful execution of our project.

1. A properly functioning workstation (PC, laptop, net-books etc.) to carry out desired research and collect relevant content.
2. Unrestricted access to the university lab in order to gather a variety of literature including academic resources (for e.g. publications, sciencedirect, journals etc.), technical manuscripts, etc. Python Idle is required to run the all the files and applications. GUI is created to interact with the user.

1.6 MOTIVATION

The development of chatbot that recommends government policies holds immense potential for enhancing the individual's confidence in applying and the willingness for them to avail the policy implemented by the government. By providing the accessible information to the Chatbot the chatbot helps to recommend the information about the relevant policies that they can apply for. It the responsibility of the government to implement the policies related to the people's healthcare, environment, economy, etc. This innovative and informative tool helps to contribute to the sustainable development of the society and making it a valuable asset to the government.

CHAPTER 2

2.1 LITRETURE SURVEY

The Chatbot Management Process is a systematic approach to managing content on chatbot systems [1], drawing from the insights gained from Evatalk, the chatbot used by the Brazilian Virtual School of Government. The methodology emphasizes the development of chatbot content by analyzing user interactions, enabling a cyclical and human-guided process. Based on the analysis of Evatalk's 21,771 contacts in 2020, the methodology successfully decreased the frequency of human intervention, enhanced the number of examples in the knowledge base, and ensured a high level of confidence in the responses, all while preserving user pleasure.

The objective of the project is to utilize artificial intelligence [2] to automate university inquiries via a web platform. This platform will enable students to engage with it by either writing or speaking in natural language. The platform will enable students to inquire about diverse subjects, including the identification of places or things, career-related matters, project themes, and graduation exercises. The platform will improve the whole university campus experience for students, teachers, and staff. The training involved utilizing 40 papers that were indexed in Scopus, which facilitated efficient communication between translators and users. The objective of the project is to streamline manual tasks carried out by university personnel by implementing a virtual assistant that can handle intricate information management. The voice-activated technology is strategically positioned around the university, enabling instantaneous engagement and enhanced reaction time. Further research is required to enhance and streamline the process of incorporating additional training materials.

Chatbots are being utilized more and more in different industries [3], including government services, with the aim of enhancing public services and cutting down on expenses. Nevertheless, the wide range of topics that fall under the purview of governmental entities poses difficulties in the development of chatbots. This study suggests utilizing text classification methods to facilitate a tree-based structure for conversational assistants. The framework enables a universal conversational AI to recognize user messages and direct them to designated specialized chatbots. Comparative experiments were conducted to evaluate the performance of multiclass, binary, and one-class classifiers. The results indicated superior performance in terms of Fl-score and accuracy metrics. Nevertheless, one-class classifiers provide significant advantages for

hierarchical chatbots because of the autonomy across models, leading to enhanced scalability and simplicity.

This study investigates the influence of interactions between chatbots and customers [4], specifically on the promotion of brands, among a sample of 312 e-commerce users who utilize chatbot systems powered by artificial intelligence. The findings indicate that meaningful human-like interactions have a major impact on brand advocacy. Additionally, consumer experiences with the brand play a role in connecting chatbot-customer interactions to brand advocacy. This underscores the increasing frequency of chatbot-customer engagement in the field of electronic commerce.

The progress in technologies such as AI, Big Data, and IoT [5] has resulted in the creation of chatbots, which are artificial intelligences that imitate human behavior and automate repetitive jobs. A study has developed a college inquiry chatbot called "College Enquiry Chatbot" that utilizes Rasa technology to evaluate and comprehend user queries. The chatbot utilizes the Rasa Core and Rasa Natural Language Understanding (NLU) packages to develop an AI chatbot that understands and responds in context. Natural Language Understanding (NLU) deduces the intention and retrieves essential elements from the user's input, whereas Rasa Core constructs a probabilistic model with a Recurrent Neural Network (RNN). The evaluation of performance criteria such as Precision, Accuracy, and F1 Score resulted in average scores of 0.628, 0.725, and 0.669, respectively. The chatbot's precision, independence from human resources, round-the-clock availability, and little upkeep make it well-suited for a range of applications, such as educational institutions and environments where querying can be burdensome. This study investigates the influence of interactions between chatbots and customers, specifically on the promotion of brands, among a sample of 312 e-commerce users who utilize chatbot systems powered by artificial intelligence. The findings indicate that the way chatbots communicate with customers in a human-like manner has a major impact on brand advocacy. Additionally, the customer's experience with the brand plays a role in connecting the chatbot-customer contact to brand advocacy. This emphasizes the increasing frequency of chatbot-customer communication in the field of electronic commerce.

AI chatbots have the potential to enhance healthcare services by emulating human-like interactions [6] and offering medical guidance. This comprehensive literature review

specifically examines the efficacy of disease prediction and the possibilities for early intervention and treatment. An extensive analysis of 24 carefully chosen academic publications, with a specific emphasis on studies published in the year 2020, revealed that AI chatbots have the potential to make a substantial impact on disease prediction and provide valuable support to healthcare professionals in making well-informed decisions. Through the application of machine learning methods and methodologies, chatbots can improve the precision and speed of disease diagnosis. Nevertheless, additional investigation and advancement are necessary to enhance the skills of AI chatbots and transform healthcare delivery to enhance patient outcomes and disease control.

The utilization of chatbots in the field of education has experienced a surge [7] in popularity as a result of technological improvements. These chatbots can assist in resolving logistical and various challenges encountered in regular classrooms. Artificial Intelligence has led to the development of platforms by digital giants such as Apple, Google, and Amazon that prioritize discussion over technical programming. This study introduces a rule-based chatbot implemented on Discord, demonstrating the integration of chatbots into various online platforms to tackle instructional difficulties. Q/A features are utilized to offer comprehensive information regarding the preexisting data within the chatbot. This technology has the potential to narrow the divide in the educational environment and offer a more captivating and efficient learning encounter.

This paper presents the Intelligent Career management Chatbot (ICMC),[8] which is specifically developed to offer career advice to students in the 10th and 12th grades. The chatbot aims to assist students in resolving their uncertainties and making informed decisions regarding their future academic pursuits. A specialized curriculum has been created specifically for students pursuing a Bachelor of Technology degree in the fields of Computer Science and Information Technology. The chatbot initiates interaction with users by posing inquiries, necessitating their response in the form of either 1 to indicate veracity or 0 to indicate falsity. The chatbot thereafter determines likely forthcoming occurrences or outcomes by analyzing the user's input and presents them on a graphical user interface (GUI). The chatbot additionally engages in communication with users. The development encompasses four primary divisions: training, incorporating features and a GUI-based chatbot server. The chatbot facilitates the process of identifying students' interests and subsequently selecting suitable career paths.

Business Intelligence (BI) is essential for firms to make informed decisions [9] based on

data analysis and visualization of past data. Nevertheless, the development of BI chatbots remains challenging, leading to ongoing problems with accessibility and usability. This study presents a model-driven methodology for the automated creation of individualized business intelligence chatbots. The approach comprises a modeling component that enables users to create a chatbot tailored to their business needs, and an automation component that generates the code for the chatbot. An empirical investigation was carried out to assess the suitability of this methodology, demonstrating encouraging outcomes for creating interactive business intelligence chatbots customized to diverse organizational requirements.

This study investigates the influence of chatbots on customer service satisfaction through the use of a quantitative review methodology [10]. The study revealed that chatbots enhance satisfaction in comparison to traditional customer service by offering convenience, effectiveness, decreased service duration, enhanced information provision, heightened productivity, and personalized encounters. The findings have a positive impact on the market as they enable the evaluation of chatbot applications, which is crucial in addressing customer happiness, a key concern in this research. In the event that a company neglects to resolve this matter, users have the option to select an alternative application offered by rival companies. The study emphasizes the significance of attending to customer happiness in order to guarantee the triumph of chatbots in the contemporary digital age.

.

2.2 PROPOSED SYSTEM

The system functions as an interactive chatbot, designed to engage users in natural language conversations. It begins with the user's input, facilitated by a graphical user interface (GUI) developed using app.py. This interface allows users to type their queries, which are then processed by the system. Upon receiving user input, the text is pre-processed using the Natural Language Toolkit (NLTK) library. NLTK provides functionalities for tokenization, which involves breaking down the input text into individual words or tokens, and stemming, which reduces words to their root form. These preprocessing steps help standardize the text and improve the model's understanding of user queries. Simultaneously, the system loads predefined intents from a JSON file named intents.json. These intents represent various topics or purposes of user queries, along with corresponding patterns of user input and appropriate responses. These intents serve as the training data for the neural network model. The training process takes place in train.py, where NLTK is utilized for data preprocessing tasks such as tokenization and stemming. PyTorch, a powerful deep learning framework, is employed to construct and train the neural network. The network architecture is designed to learn from the input patterns and associate them with the appropriate intents, allowing it to generate relevant responses during conversations. During training, the model iterates through the dataset of intents, adjusting its parameters to minimize the prediction errors. Once the training is complete, the model's parameters are saved in a file named data.pytorch. This file contains the learned weights and biases of the neural network, enabling efficient loading for subsequent interactions. In the chat interface (chat.py), the trained model is loaded to process user inputs in real-time. The model predicts the intent of the user query based on the learned patterns and generates an appropriate response accordingly. These responses are then displayed to the user through the GUI, creating a seamless conversational experience. Overall, the system's architecture leverages NLTK for text preprocessing, PyTorch for neural network training, and a GUI developed using Tkinter for user interaction. By combining these technologies, the system provides an intuitive and engaging chatbot interface capable of understanding and responding to user queries effectively.

2.3 NLTK FRAMEWORK

NLTK (Natural Language Toolkit) is a comprehensive library for natural language processing (NLP) and text analysis in Python. It provides tools and resources for a wide range of NLP tasks, including tokenization, part-of-speech tagging, named entity recognition, sentiment analysis, machine translation, and more. Here's an overview of some key aspects of NLTK. NLTK offers various tokenization methods for splitting text into words or sentences. This includes basic tokenization as well as more advanced methods like word tokenization, sentence tokenization, and regular expression-based

tokenization. NLTK provides tools for tagging words in a text with their corresponding parts of speech, such as nouns, verbs, adjectives, and adverbs. It includes pre-trained models for POS tagging in multiple languages. NLTK includes tools for identifying and extracting named entities from text, such as names of people, organizations, locations, dates, and numerical expressions. It uses machine learning algorithms to recognize named entities in unstructured text data. NLTK supports parsing of syntactic structures in sentences, including constituency parsing and dependency parsing. It allows you to analyze the grammatical structure of sentences and extract information about relationships between words. NLTK supports machine translation tasks by providing interfaces to external translation services and tools. It allows you to translate text between different languages using pre-trained translation models or online translation services.

2.4 MACHINE LEARNING

Neural Network Architecture

The neural network has the following architecture:

- Input Layer: The size of the input layer is determined by the input_size parameter.
- Hidden Layers: There are two hidden layers, each with hidden_size neurons. These layers perform transformations on the input data.
- Output Layer: The output layer has num_classes neurons, representing the number of classes in the classification task.

Activation Function

The ReLU (Rectified Linear Unit) activation function is used between the layers. ReLU introduces non-linearity into the network, allowing it to learn and model complex relationships in the data. It replaces negative values with zero and leaves positive values unchanged.

Forward Pass

The data is passed through different layers. The output of one layer serves as the input to the next layer, and this process continues until the data passes through all the layers of the network. Finally, the output layer produces the network's prediction, which is compared to the actual target value during training to compute the loss and update the network's parameters through backpropagation.

2.5 GUI CREATION

In app.py, the GUI for the chatbot application is created using the Tkinter framework. This script defines the main window's configuration, including its title, size, and background color. It sets up the layout by adding labels, text widgets, entry fields, buttons, etc. Additionally, it implements event handlers to manage user interactions such as sending messages and pressing the Enter key. The main application entry point checks if the script is being run as the main program and starts the chat application accordingly. This script integrates the chatbot's functionality from other modules, such as chat.py, to provide users with a user-friendly interface for interacting with the chatbot.

CHAPTER 3

SYSTEM DESIGN

3.1 GENERAL

In this section, we would like to show how the general outline of how all the components end up working when organized and arranged together. It is further represented in the form of a flow chart below.

3.2 SYSTEM ARCHITECTURE DIAGRAM

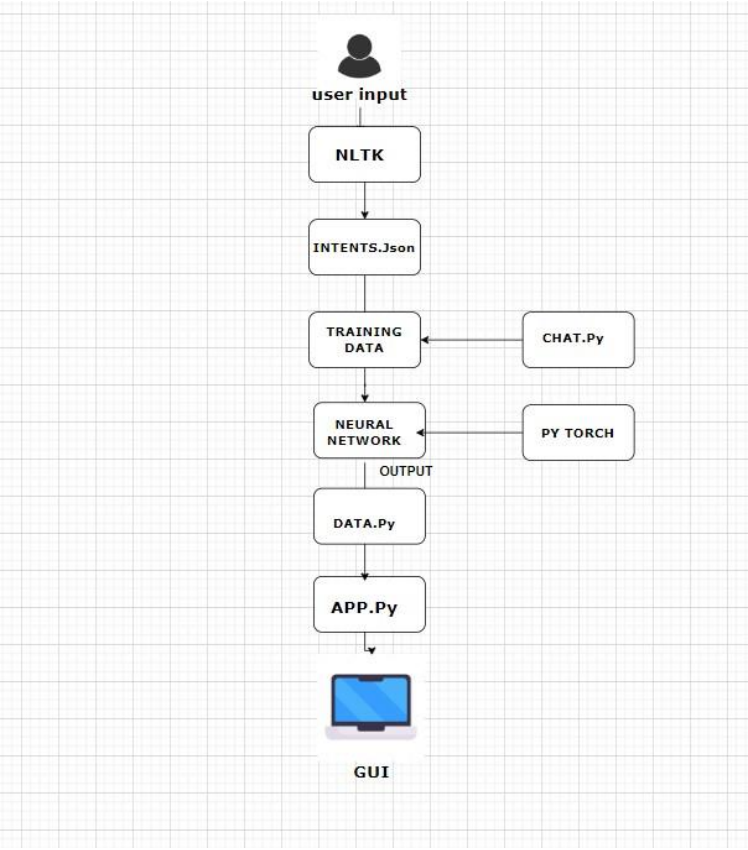


Fig 3.1: System Architecture

3.3 DEVELOPMENTAL ENVIRONMENT

3.3.1HARDWARE REQUIREMENTS

The hardware requirements may serve as the basis for a contract for the system's implementation. It should therefore be a complete and consistent specification of the entire system. It is generally used by software engineers as the starting point for the system design.

Table 3.1 Hardware Requirements

COMPONENTS	SPECIFICATION
PROCESSOR	Intel Core i5
RAM	8 GB RAM
GPU	NVIDIA GeForce GTX 1650
MONITOR	15" COLOR
HARD DISK	512 GB
PROCESSOR SPEED	MINIMUM 1.1 GHz

3.3.2SOFTWARE REQUIREMENTS

The software requirements document is the specifications of the system. It should include both a definition and a specification of requirements. It is aset of what the system should rather be doing than focus on how it should be done. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating the cost, planning team activities, performing tasks, tracking the team, and tracking the team's progress throughout the development activity.

It includes the **Python IDLE version 3.12, Chrome, and Command Prompt**. The necessary libraries for the execution of the Chat Bot are installed using the idle and the Command prompt.

CHAPTER 4

PROJECT DESCRIPTION

4.1 METHODOLODGY

In the development of the of a chat bot it includes gathering the information regarding all the Government schemes related to the Health. The NLP provides essential functions for text preprocessing in natural language processing (NLP) tasks. It includes functions for tokenization, stemming, and creating a bag of words representation. Tokenization splits a sentence into individual words or tokens. Stemming reduces words to their base or root form to standardize them. The bag of words representation converts a sentence into a binary vector indicating the presence or absence of each word from a predefined set of words. These preprocessing steps are crucial for preparing text data for various NLP applications, such as sentiment analysis, text classification, and information retrieval.

The JSON structure represents a collection of intents and their corresponding patterns and responses, typically used in chatbot development. Each intent represents a specific topic or purpose, such as greetings, farewells, thanks, or inquiries about products/services. Patterns are example phrases or questions that users might input, while responses are the bot's replies to those patterns. This structure enables the chatbot to understand user queries and provide appropriate responses, making interactions more conversational and intuitive. Additionally, it allows for easy management and customization of the bot's behaviour and conversation flow.

The train.py is responsible for training the neural network model by processing the dataset of intents and responses, preparing the data for training, and saving the trained model. model.py defines the architecture of the neural network model used in the chatbot, specifying its layers and activation functions. chat.py utilizes the trained model to process user input, generate responses, and facilitate the conversation with users. Together, these files form the backbone of the chatbot system, enabling it to understand user queries and provide relevant answers.

The app.py is the main script responsible for creating the graphical user interface (GUI) of the chatbot application using the Tkinter library in Python. It defines a class called ChatApplication that sets up the main window, including labels, text widgets, entry boxes, and buttons for sending messages. The class also handles user interactions, such as sending messages and displaying responses, by binding functions to GUI events. Overall, app.py serves as the entry point for launching the chatbot application and provides the user interface for interacting with the chatbot.

4.2 MODULE DESCRIPTION

The development of an AI-driven Chatbot includes several module each performing the hey roles in achieving the overall user input and response of chatbot through GUI.

4.2.1 NLTK MODULE:

The NLTK (Natural Language Toolkit) module to performs text preprocessing tasks. It includes functions for tokenization, stemming, and creating a bag-of-words representation of text data. Tokenization breaks down sentences into individual words or tokens, while stemming reduces words to their root form. The bag of words function converts tokenized sentences into numerical vectors, where each element represents the presence or absence of a word from a predefined vocabulary. This preprocessing is fundamental for various natural language processing tasks, such as text classification and sentiment analysis, as it transforms raw text data into a format suitable for machine learning algorithms.

4.2.2 JSON MODULE

The JSON data represents a collection of intents for a chatbot or virtual assistant. Each intent contains a tag, which serves as an identifier, a set of patterns representing user inputs or queries, and corresponding responses that the chatbot can provide. For example, there are intents for greeting, saying goodbye, expressing thanks, asking about available items or payment methods, inquiring about delivery times, requesting jokes, and seeking information on various government schemes and health programs. Each pattern is a potential user query, and the responses are appropriate replies generated by the chatbot based on the detected intent. This structured data is commonly used to train and configure conversational AI systems.

4.2.3 TRAINING MODULE

The training module processes intents from a JSON file, preparing data for model training using techniques like tokenization and bag-of-words. It then trains a neural network model on this data, teaching it to classify input queries and generate appropriate responses. On the other hand, chat.py utilizes the trained model to create a conversational interface, interacting with users in real-time. It tokenizes user input, feeds it to the model for prediction, and generates responses based on the model's output. Together, these scripts form the backbone

of a chatbot system, enabling it to understand user queries and provide relevant responses effectively.

4.2.4 NEURAL NETWORK MODULE

The PyTorch library is utilized to define a neural network architecture for the chatbot. This architecture typically includes layers for input processing, hidden representations, and output classification. The model is trained using the data prepared in train.py and stored in a PyTorch data file (usually with the .pth extension), which contains information such as model parameters and vocabulary. During inference, chat.py loads this trained model from the data file to generate responses to user queries based on the learned patterns and associations. Overall, PyTorch facilitates the creation, training, and deployment of deep learning models, enabling efficient processing of natural language data for chatbot applications.

4.2.5 GUI MODULE

In app.py, a graphical user interface (GUI) is created using the Tkinter library in Python. The GUI provides a user-friendly interface for interacting with the chatbot. The main window of the application is set up with specific dimensions, title, and resizable options. Within the window, various GUI elements are organized, including labels for displaying information, a text widget for showing the conversation history, an entry widget for typing messages, and a button for sending messages. Scrollbars are added to allow scrolling through the conversation history if needed. Overall, app.py orchestrates the creation and layout of these GUI components to deliver a seamless chatbot user experience.

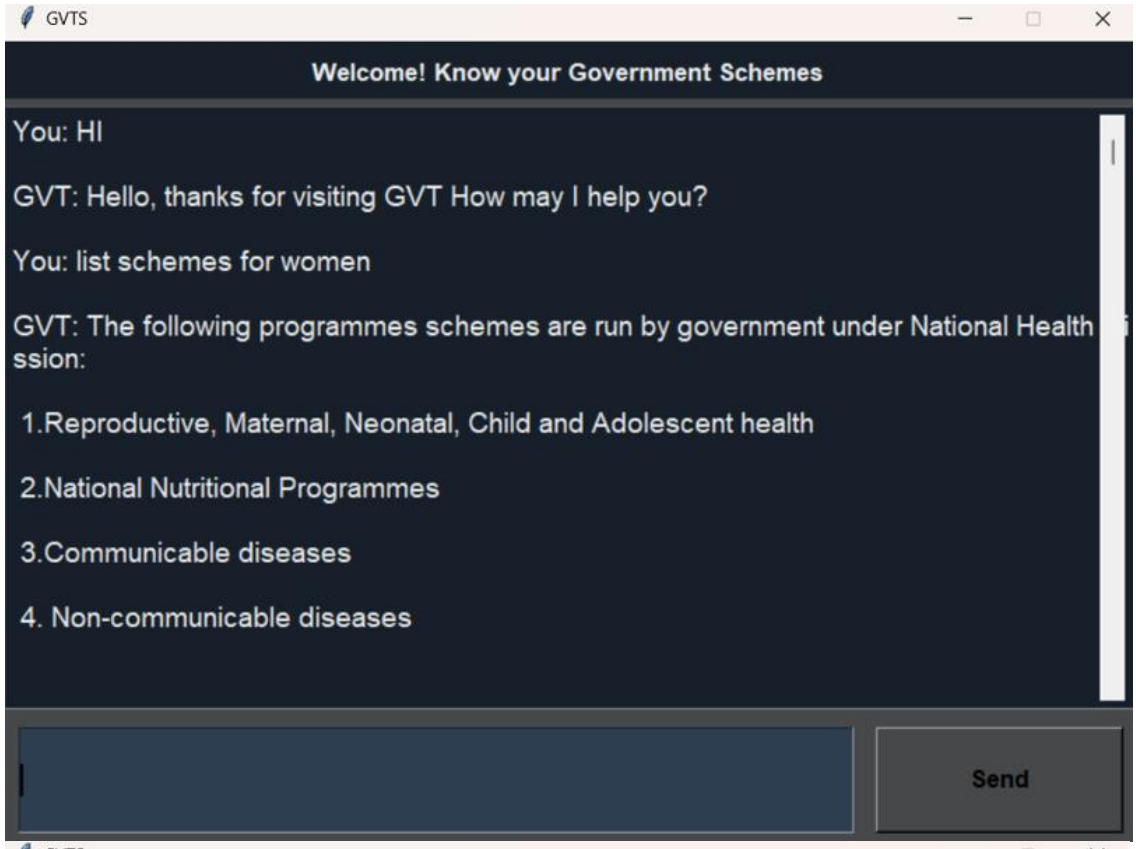
.

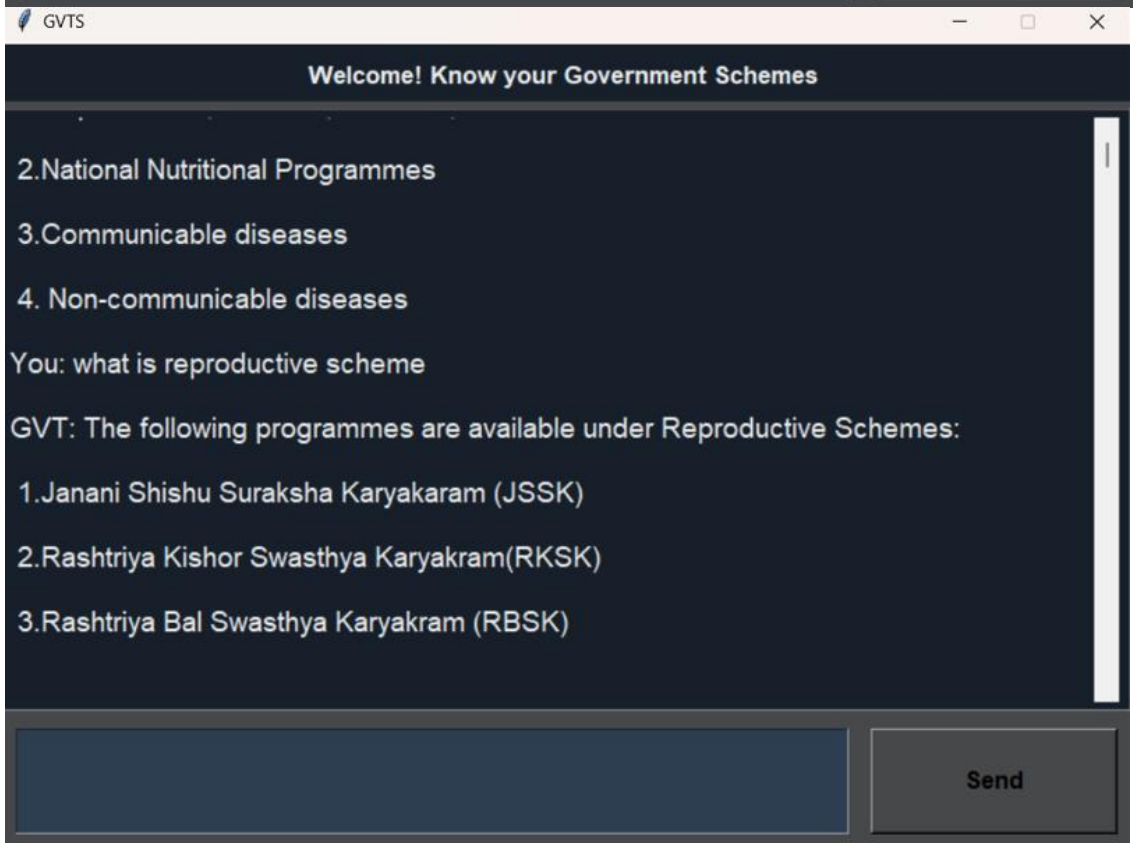
CHAPTER 5
RESULTS AND DISCUSSIONS

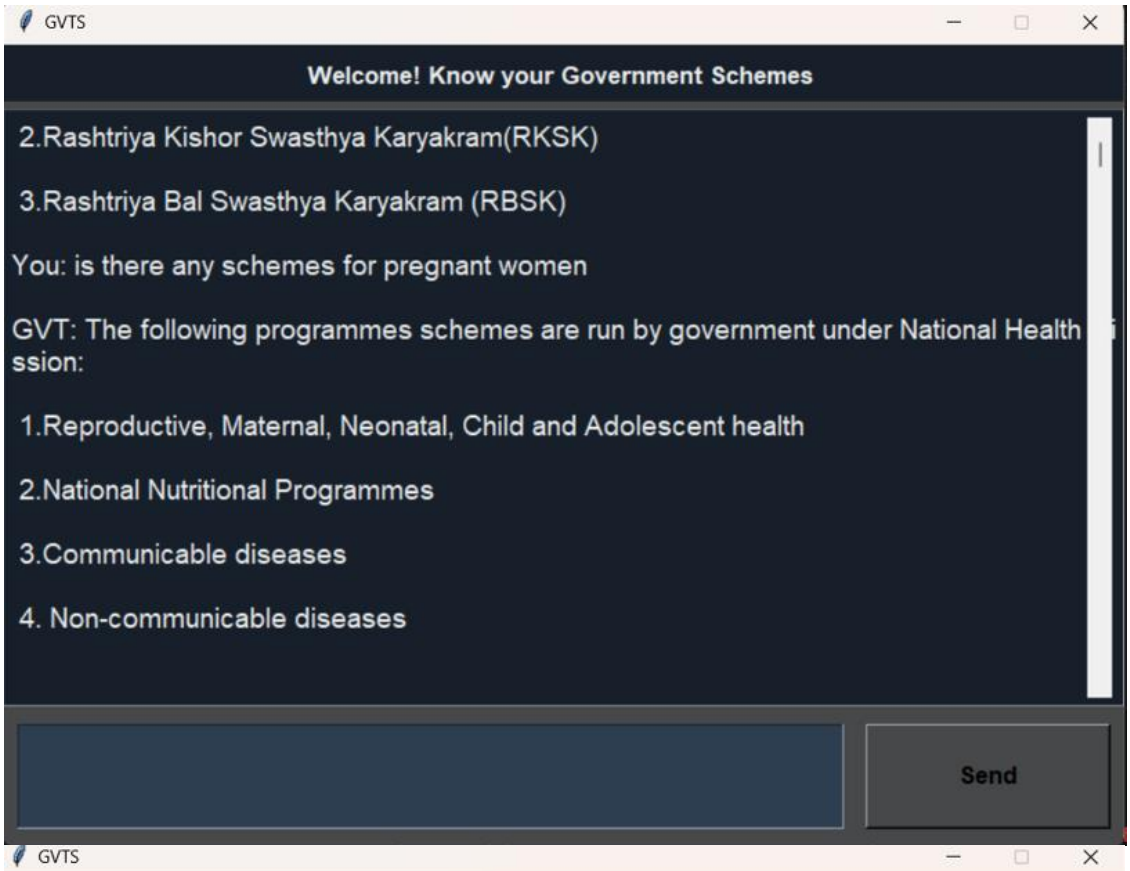
5.1 OUTPUT

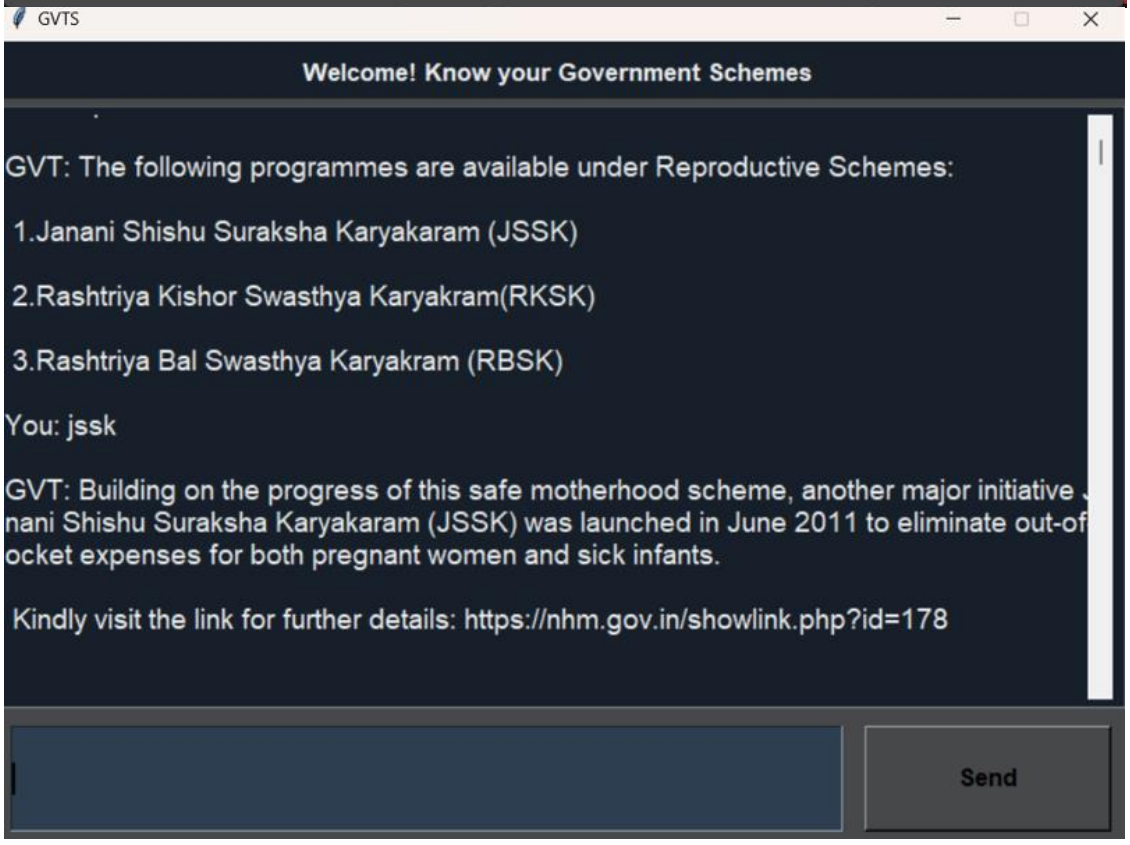
The following images contain images attached below of the working application.

Example instance of creating a generation


GVTS
Welcome! Know your Government Schemes
You: HI
GVT: Hello, thanks for visiting GVT How may I help you?
You: list schemes for women
GVT: The following programmes schemes are run by government under National Health ssion:
1.Reproductive, Maternal, Neonatal, Child and Adolescent health
2.National Nutritional Programmes
3.Communicable diseases
4. Non-communicable diseases
Send


GVTS
Welcome! Know your Government Schemes
2.National Nutritional Programmes
3.Communicable diseases
4. Non-communicable diseases
You: what is reproductive scheme
GVT: The following programmes are available under Reproductive Schemes:
1.Janani Shishu Suraksha Karyakaram (JSSK)
2.Rashtriya Kishor Swasthya Karyakram(RKSK)
3.Rashtriya Bal Swasthya Karyakram (RBSK)
Send


GVTS
Welcome! Know your Government Schemes
2.Rashtriya Kishor Swasthya Karyakram(RKSK)
3.Rashtriya Bal Swasthya Karyakram (RBSK)
You: is there any schemes for pregnant women
GVT: The following programmes schemes are run by government under National Health
ssion:
1.Reproductive, Maternal, Neonatal, Child and Adolescent health
2.National Nutritional Programmes
3.Communicable diseases
4. Non-communicable diseases
Send


GVTS
Welcome! Know your Government Schemes
GVT: The following programmes are available under Reproductive Schemes:
1.Janani Shishu Suraksha Karyakaram (JSSK)
2.Rashtriya Kishor Swasthya Karyakram(RKSK)
3.Rashtriya Bal Swasthya Karyakram (RBSK)
You: jssk
GVT: Building on the progress of this safe motherhood scheme, another major initiative
nani Shishu Suraksha Karyakaram (JSSK) was launched in June 2011 to eliminate out-of
ocket expenses for both pregnant women and sick infants.
Kindly visit the link for further details: https://nhm.gov.in/showlink.php?id=178
Send

5.2 RESULT

The proposed system is an interactive chatbot application aimed at assisting users in accessing information about various government schemes and programs. It leverages natural language processing techniques, specifically using the NLTK library for text processing and PyTorch for training a neural network model. The system begins by collecting user input, which is then tokenized and stemmed to extract meaningful information. Intents for different user queries are defined in a JSON file, which is used for training the chatbot model. The training process involves mapping user inputs to predefined intents and responses. Once trained, the model is capable of recognizing user queries and providing appropriate responses based on the identified intent. Additionally, the application includes a graphical user interface (GUI) created using Tkinter, allowing users to interact with the chatbot seamlessly. The trained model is then integrated into the GUI, enabling users to input queries and receive accurate and informative responses about government schemes and programs. Overall, the system aims to enhance accessibility to government initiatives and promote public awareness of available services.

CHAPTER 6

CONCLUSION AND FUTURE ENHANCEMENT

6.1 CONCLUSION

The proposed chatbot system represents a significant step towards enhancing accessibility to government schemes and programs. By leveraging natural language processing techniques and machine learning, the system effectively interprets user queries and provides relevant information in real-time. The inclusion of a graphical user interface (GUI) ensures user-friendly interaction, making the system accessible to a wide range of users. Furthermore, the integration of PyTorch for training a neural network model enables accurate recognition of user intents, thereby improving the quality of responses. Overall, this system serves as a valuable tool for individuals seeking information about government initiatives, empowering them to make informed decisions and utilize available resources effectively. As technology continues to advance, further enhancements and refinements to the system can be made, ultimately contributing to greater transparency, efficiency, and citizen engagement in governance.

6.2 FUTURE ENHANCEMENT

Incorporating a feedback mechanism where users can rate the helpfulness and accuracy of responses can provide valuable insights for continuously improving the system. By analyzing user feedback, the system can identify areas for improvement and refine its responses over time.

Moreover, enhancing the system's ability to handle multilingual queries can broaden its accessibility to users from diverse linguistic backgrounds. Implementing language detection algorithms and integrating translation services could facilitate seamless communication with users in their preferred language.

APPENDIX

SOURCE CODE:

intents.py :

```
{ "intents": [
{
  "tag": "greeting",
  "patterns": [
    "Hi",
    "Hey",
    "How are you",
    "Is anyone there?", ],
  "responses": [
    "Hi Welcome to GVT How May I help you?",
    "Hello, thanks for visiting GVT How may I help you?",
    "Hi there, what can I do for you?",
    "Hi there, how can I help?"
  ]
},{
  "tag": "goodbye",
  "patterns": ["Bye", "See you later", "Goodbye"],
  "responses": [
    "See you later, thanks for visiting",
    "Have a nice day",
    "Bye! Come back again soon."
  ]
},{
```

```
  "tag": "thanks",
  "patterns": ["Thanks", "Thank you", "That's helpful", "Thank's a lot!"],
  "responses": ["Happy to help!", "Any time!", "My pleasure"]
},{
  "tag": "items",
  "patterns": [
    "Which items do you have?",
    "What kinds of items are there?",
    "What do you sell?"
  ],
  "responses": [
    "We sell coffee and tea",
    "We have coffee and tea"
  ]
},{
  "tag": "payments",
  "patterns": [
    "Do you take credit cards?",
    "Do you accept Mastercard?",
    "Can I pay with Paypal?",
    "Are you cash only?"
  ],
  "responses": [
    "We accept VISA, Mastercard and Paypal",
    "We accept most major credit cards, and Paypal"
  ]
},{
```

```
 "tag": "delivery",
 "patterns": [
  "How long does delivery take?",
  "How long does shipping take?",
  "When do I get my delivery?"
 ],
 "responses": [
  "Delivery takes 2-4 days",
  "Shipping takes 2-4 days"
 ]}, {
 "tag": "funny",
 "patterns": [
  "Tell me a joke!",
  "Tell me something funny!",
  "Do you know a joke?"
 ],
 "responses": [
  "Why did the hipster burn his mouth? He drank the coffee before it was cool.",
  "What did the buffalo say when his son left for college? Bison."
 ]
},{
 "tag": "message",
 "patterns": [
  "may I know the schemes available for women in India",
  "Schemes for women",
  "Women Schemes",
  "What schemes available for women in india",
```

```
    "Any govenment schemes for women",

    "Women Entreprenuer"

  ],

  "responses": [

    "The following programmes schemes are run by government under National Health Mission: \n\n 1.Reproductive, Maternal, Neonatal, Child and Adolescent health  \n\n 2.National Nutritional Programmes \n\n 3.Communicable diseases \n\n 4. Non-communicable diseases"]

}, {

  "tag": "Reproductive",

  "patterns": [

    "Reproductive, Maternal, Neonatal, Child",

    "reproductive",

    "Neonatal",

    "Child",

    "maternal",

    "Adoloscent Health",

    "What is Reproductive Scheme",

    "What is Neonatal Scheme"

  ],

  "responses": [

    "The following programmes are available under Reproductive Schemes: \n\n 1.Janani Shishu Suraksha Karyakaram (JSSK)  \n\n 2.Rashtriya Kishor Swasthya Karyakram(RKSK) \n\n 3.Rashtriya Bal Swasthya Karyakram (RBSK)  "]{

  "tag": "Janani Shishu Suraksha Karyakaram (JSSK)",

  "patterns": [

    "Janani Shishu Suraksha Karyakaram (JSSK)",

    "JSSK",

    "Explain about JSSK scheme",

    "what is JSSK Scheme"
```

```
  ],

  "responses": [

    "Building on the progress of this safe motherhood scheme, another major initiative Janani Shishu Suraksha Karyakaram (JSSK) was launched in June 2011 to eliminate out-of-pocket expenses for both pregnant women and sick infants. \n\n Kindly visit the link for further details: https://nhm.gov.in/showlink.php?id=178"

  ]

},

{

  "tag": "Rashtriya Kishor Swasthya Karyakram(RKSK)",

  "patterns": [

    "Rashtriya Kishor Swasthya Karyakram(RKSK)",

    "RKSK",

    "Explain about RKSK scheme",

    "what is RKSK Scheme"

  ],

  "responses": [

    "The Rashtriya Kishor Swasthya Karyakram was launched on 7th January, 2014. The key principle of this programme is adolescent participation and leadership, Equity and inclusion, Gender Equity and strategic partnerships with other sectors and stakeholders. \n\n Kindly visit the link for further details: https://rksk.in/home"

  ]

},

{

  "tag": "Rashtriya Bal Swasthya Karyakram (RBSK)",

  "patterns": [

    "Rashtriya Bal Swasthya Karyakram (RBSK)",

    "RBSK",
```

```
    "Explain about RBSK scheme",

    "what is RBSK Scheme"

  ],

  "responses": [

    "The Ministry of Health & Family Welfare, Government of India, under the National Health Mission launched the Rashtriya Bal Swasthya Karyakram (RBSK), an innovative and ambitious initiative, which envisages Child Health Screening and Early Intervention Services, a systemic approach of early identification and link to care, support and treatment. \n\n Kindly visit the link for further details: https://rbsk.mohfw.gov.in/RBSK/"



  ]

},

{

  "tag": "National Iodine Deficiency Disorders Control Programme ",

  "patterns": [

    "National Iodine Deficiency Disorders Control Programme ",

    "NIDDCP",

    "Iodine Deficiency",

    "what is Iodine Deficiency Scheme"

  ],

  "responses": [

    "The Ministry of Health & Family Welfare, Government of India, under the National Health Mission launched the Rashtriya Bal Swasthya Karyakram (RBSK), an innovative and ambitious initiative, which envisages Child Health Screening and Early Intervention Services, a systemic approach of early identification and link to care, support and treatment. \n\n Kindly visit the link for further details:https://nhm.gov.in/index1.php?lang=1&level=3&sublinkid=1054&lid=230"]

{

  "tag": "Thank you",

  "patterns": [

    "Thanks",

    "Thankyou",

    "Thanks for your Help"
```

```
    ],
    "responses": [
      "Welcome"
    ]}
```

model.py

```
import torch
import torch.nn as nn
class NeuralNet(nn.Module):
    def __init__(self, input_size, hidden_size, num_classes):
        super(NeuralNet, self).__init__()
        self.l1 = nn.Linear(input_size, hidden_size)
        self.l2 = nn.Linear(hidden_size, hidden_size)
        self.l3 = nn.Linear(hidden_size, num_classes)
        self.relu = nn.ReLU()
    def forward(self, x):
        out = self.l1(x)
        out = self.relu(out)
        out = self.l2(out)
        out = self.relu(out)
        out = self.l3(out)
        # no activation and no softmax at the end
        return out
```

app.py

```
 from tkinter import *
from chat import get_response, bot_name
BG_GRAY = "#46484a"
BG_COLOR = "#17202A"
```

```
TEXT_COLOR = "#EAECEE"
FONT = "Helvetica 14"
FONT_BOLD = "Helvetica 13 bold"
class ChatApplication:
    def __init__(self):
        self.window = Tk()
        self._setup_main_window()

    def run(self):
        self.window.mainloop()

    def _setup_main_window(self):
        self.window.title("GVTS")
        self.window.resizable(width=False, height=False)
        self.window.configure(width=770, height=550, bg=BG_COLOR)

        head_label = Label(self.window, bg=BG_COLOR, fg=TEXT_COLOR,
                           text="Welcome! Know your Government Schemes ", font=FONT_BOLD, pady=10)
        head_label.place(relwidth=1)

        line = Label(self.window, width=450, bg=BG_GRAY)
        line.place(relwidth=1, rely=0.07, relheight=0.012)

        self.text_widget = Text(self.window, width=20, height=2, bg=BG_COLOR, fg=TEXT_COLOR,
                                font=FONT, padx=5, pady=5)
        self.text_widget.place(relheight=0.745, relwidth=1, rely=0.08)
        self.text_widget.configure(cursor="arrow", state=DISABLED)

        scrollbar = Scrollbar(self.text_widget)
        scrollbar.place(relheight=1, relx=0.974)
        scrollbar.configure(command=self.text_widget.yview)

        bottom_label = Label(self.window, bg=BG_GRAY, height=80)
        bottom_label.place(relwidth=1, rely=0.825)

        self.msg_entry = Entry(bottom_label, bg="#2C3E50", fg=TEXT_COLOR, font=FONT)
        self.msg_entry.place(relwidth=0.74, relheight=0.06, rely=0.008, relx=0.011)
        self.msg_entry.focus()
        self.msg_entry.bind("<Return>", self._on_enter_pressed)

        send_button = Button(bottom_label, text="Send", font=FONT_BOLD, width=20, bg=BG_GRAY,
                             command=lambda: self._on_enter_pressed(None))
        send_button.place(relx=0.77, rely=0.008, relheight=0.06, relwidth=0.22)

    def _on_enter_pressed(self, event):
        msg = self.msg_entry.get()
        self._insert_message(msg, "You")

    def _insert_message(self, msg, sender):
        if not msg:
            return

        self.msg_entry.delete(0, END)
        msg1 = f"{sender}: {msg}\n\n"
```

```
        self.text_widget.configure(state=NORMAL)
        self.text_widget.insert(END, msg1)
        self.text_widget.configure(state=DISABLED)

        msg2 = f"{bot_name}: {get_response(msg)}\n\n"
        self.text_widget.configure(state=NORMAL)
        self.text_widget.insert(END, msg2)
        self.text_widget.configure(state=DISABLED)

        self.text_widget.see(END)
if __name__ == "__main__":
    app = ChatApplication()
    app.run()
```

REFERENCES

1. G. A. Santos, G. G. de Andrade, G. R. S. Silva, F. C. M. Duarte, J. P. J. D. Costa and R. T. de Sousa, "A Conversation-Driven Approach for Chatbot Management," in IEEE Access, vol. 10, pp. 8474-8486, 2022, doi: 10.1109/ACCESS.2022.3143323.

2. J. R. Caballero Castellanos, M. Cardona and J. L. Ordoñez-Avila, "Chatbot for Validation of Research Topics for Engineering Students," 2023 IEEE 41st Central America and Panama Convention (CONCAPAN XLI), Tegucigalpa, Honduras, 2023, pp. 1-5, doi: 10.1109/CONCAPANXLI59599.2023.10517536.

3. R. Agra, B. G. Resende, C. C. Wermelinger, P. C. O. Dias and M. Ladeira, "A tree-organized chatbot proposal to provide a single digital channel to access specific chatbots in a real Brazilian digital government environment," 2023 18th Iberian Conference on Information Systems and Technologies (CISTI), Aveiro, Portugal, 2023, pp. 1-6, doi: 10.23919/CISTI58278.2023.10211721.

4. Y. Liu, X. Li and Z. Xiang, "The Effect of Chatbot-customer Interaction on Consumer Brand Advocacy: Exploring the Role of Chatbots," 2022 IEEE 12th International Conference on Electronics Information and Emergency Communication (ICEIEC), Beijing, China, 2022, pp. 185-190, doi: 10.1109/ICEIEC54567.2022.9835050.

5. S. Meshram, N. Naik, M. VR, T. More and S. Kharche, "College Enquiry Chatbot using Rasa Framework," 2021 Asian Conference on Innovation in Technology (ASIANCON), PUNE, India, 2021, pp. 1-8, doi: 10.1109/ASIANCON51346.2021.9544650.

6. V. Velasco, K. Dedy Setiawan, R. Robert Sanjaya, M. Susan Anggreainy and A. Kurniawan, "AI Chatbot Technology to Predict Disease: A Systematic Literature Review," 2023 4th International Conference on Artificial Intelligence and Data Sciences (AiDAS), IPOH, Malaysia, 2023, pp. 97-101, doi: 10.1109/AiDAS60501.2023.10284717.

7. M. G. C. P, A. Srivastava, S. Chakraborty, A. Ghosh and H. Raj, "Development of Information Technology Telecom Chatbot: An Artificial Intelligence and Machine Learning Approach," 2021 2nd International Conference on Intelligent Engineering and Management (ICIEM), London, United Kingdom, 2021, pp. 216-221, doi: 10.1109/ICIEM51511.2021.9445354.

8. R. Goyal, N. Chaudhary and M. Singh, "Machine Learning based Intelligent Career Counselling Chatbot (ICCC)," 2023 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2023, pp. 1-8, doi: 10.1109/ICCCI56745.2023.10128305.

9. M. Banisharif, A. Mazloumzadeh, M. Sharbaf and B. Zamani, "Automatic Generation of Business Intelligence Chatbot for Organizations," 2022 27th International Computer Conference, Computer Society of Iran (CSICC), Tehran, Iran, Islamic Republic of, 2022, pp. 1-5, doi: 10.1109/CSICC55295.2022.9780490.

10. E. R. Putra Antonio, M. F. Fadhilah, F. Faiq, R. Fredyan and H. Pranoto, "Analyzing the Impact of Customer Service Chatbots on User Satisfaction," 2023 15th International Congress on Advanced Applied Informatics Winter (IIAI-AAI-Winter), Bali, Indonesia, 2023, pp. 82-85, doi: 10.1109/IIAI-AAI-Winter61682.2023.00023.