IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

Anglips Human Interacting Chabot Using Nlp And Deep Learning

[1]M/s Ipsita Ankita Hota, [2]Dr. Ranjan Kumar Dash

[1]Asst.Professor CVRP, [2]Proffesor Odisha University of Technology and Research
[1]Computer Science and Engineering,
[1]C.V Raman Polytechnic, Bhubaneswar, India

***Abstract**:* Anglips is a human interacting deep-learning NLP and neural network-based Chabot. It includes GUI with the help of Pyqt5 and the Qt-designer tool. Chabot is like a good talkable friend for every introverted person. Here the Chabot is designed for human interaction with both speech and text-based. Initially, a GUI displays with two buttons START and EXIT buttons by pressing the START button it will start the application. Here a speech-reorganization API takes a vital role in human interaction, it takes the voice input of the user through the microphone and converts it into machine-understandable language during which it goes from listening to recognizing mode after which the system gives a corresponding response to the user. In the proposed method 100 intents were taken with more than 4 responses for training and testing with 21 intents and gives accuracy of 95.24% which makes it more accurate. Chatbots are useful for conversation as well as business, marketing, FAQ, and entertainment purposes. Anglips supports both speech and text-based conversation.

***Index Terms* -** Deep-learning, Neural-network, Qt-designer, PyQt5, Speech-reorganization API.

1. Introduction

Chatbots, also known as conversational agents or virtual assistants, are computer programs designed to simulate human conversation through artificial intelligence (AI) and natural language processing (NLP) technologies, deep learning, neural networks, etc. They have emerged as valuable tools for businesses seeking to automate customer interactions, improve operational efficiency, and deliver personalized experiences. Beyond their traditional roles in business and FAQ support, chatbots serve as invaluable companions for conversation and anxiety relief. With their ability to engage users in light-hearted banter, share jokes, offer cute compliments, and provide empathetic responses, chatbots play a significant role in entertaining individuals and making them feel understood and supported.

This multifaceted capability not only enhances user experience but also fosters a sense of connection and reliability, making chatbots a versatile tool for improving mental well-being and social interaction. In today's digital landscape, chatbots seamlessly integrate into various domains to provide users with prompt responses to their inquiries. Operating within specific domains, chatbots serve as reliable sources of information, catering to user queries effectively and efficiently. Whether it's customer service, healthcare, e-commerce, or any other field, chatbots play a pivotal role in delivering accurate answers and assistance to users, enhancing their overall experience [1].

At their core, chatbots are computer programs designed to understand and respond to natural language input from users. They leverage advanced techniques in natural language processing (NLP) and machine learning to interpret user queries and generate appropriate responses. With each interaction, chatbots learn and adapt, improving their understanding and becoming more efficient at providing helpful information. Several key trends have fueled the rise of chatbots. The proliferation of smartphones and other mobile devices has made it easier than ever to access digital services on the go. At the same time, the growing availability of data and the increasing power of AI algorithms have enabled more sophisticated chatbot designs. Additionally, a greater emphasis on user-centric design has led to a demand for technology that feels intuitive and personal. Chatbots offer several advantages over traditional methods of interacting with technology. They provide instant, 24/7 availability, eliminating the need for users to wait for responses. Chatbots can handle multiple conversations simultaneously, making them highly efficient. Additionally, they can be programmed to access and retrieve information from vast databases, providing users with quick and accurate answers. Despite their growing popularity, chatbots are still a relatively new technology, and challenges remain. Accuracy in understanding user input can be inconsistent, and generating natural and engaging responses is a complex problem. Privacy and security concerns are also paramount, as chatbots often handle sensitive user data. As chatbots continue to evolve, they have the potential to revolutionize how we interact with technology. By simulating human conversation in a more lifelike manner, they can make digital systems feel more intuitive and personal. In the future, chatbots may even become our primary interface for interacting with the digital world, serving as our digital assistants.

2. Literature Survey

Authors [1] have introduced a chatbot using Rasa, highlighting its effectiveness in accurately identifying user intents. They noted that future enhancements would include adding speech-to-text capabilities. This feature aims to provide better entertainment options for users experiencing stress. In their work, Author [2] has designed a text-based chatbot that utilizes pattern matching to recognize user inputs. Authors [3] created a chatbot to efficiently address student inquiries, leveraging natural language processing (NLP) and deep learning techniques. Authors [4] developed a chatbot utilizing deep neural networks (DNN) and a graphical user interface (GUI) created with Tkinter. This chatbot is capable of delivering responses and displaying facial attributes, enhancing user interaction. By incorporating advanced AI techniques and intuitive design, these chatbots aim to improve user experience through sophisticated response generation and interactive elements. The implementation of DNN and Tkinter by authors [4] brings a unique dimension to chatbot interactions, allowing for visual feedback alongside text-based communication. Additionally, authors [5] implemented a chatbot that integrates artificial intelligence (AI) and natural language processing (NLP). This system is designed with a user-friendly GUI featuring various widgets, facilitating seamless interaction. It is specifically tailored to address queries related to examinations, academic subjects, and other educational concerns, providing a comprehensive support tool for students. Meanwhile, the work by authors [5] ensures that the chatbot serves as a valuable academic resource, effectively utilizing NLP to understand and respond to a wide range of educational queries. Together, these advancements demonstrate the potential of AI-driven chatbots in educational and user support applications. The chatbot system proposed by the author [6] is designed to help students stay informed and easily access college-related information. It processes input from students and provides relevant answers, making it a valuable tool for managing inquiries about college activities and updates. Overall, these advancements in chatbot technology demonstrate significant potential in educational environments. The system proposed by the authors [6] focuses on maintaining student engagement with up-to-date college information. Overall, these advancements in chatbot technology demonstrate significant potential in educational environments. The system proposed by the authors [6] focuses on maintaining student engagement with up-to-date college information. Authors [7] developed a college inquiry chatbot that leverages AI and natural language processing (NLP). This chatbot allows users to ask various questions related to college, ensuring they receive accurate and timely information. The AI and NLP-based inquiry chatbot by the authors [7] serves as a comprehensive tool for addressing college-related questions. In another development, the authors [8] showcased a chatbot built using deep learning techniques, specifically employing bidirectional recurrent neural networks (RNNs). This approach enhances the chatbot's ability to understand and generate responses based on the context of the conversation.

Furthermore, authors [9] implemented a university FAQ chatbot using deep learning, achieving an impressive accuracy rate of 91.55%. This system is currently text-based, but future enhancements will include voice recognition capabilities to improve user interaction and accessibility. Authors [10] utilized Artificial Intelligence Markup Language (AIML) and Latent Semantic Analysis (LSA) in their chatbot methodology. This approach is designed to facilitate interactions within any university setting, providing a robust framework for handling a variety of user queries, the methodology by authors [10] using AIML and LSA provides a versatile solution for university interactions, capable of adapting to various inquiry types. These innovations collectively enhance the efficiency and effectiveness of chatbots in educational contexts, offering valuable support to students and institutions alike.

3. Proposed Anglips Chatbot System

The main aim of the Anglips project is to create a user-friendly interface that encourages open and honest communication. The goal is to provide a safe space where people can freely express their feelings and thoughts, as they would with a trusted friend.

3.1. Data Set:

A JSON file is a type of file format used for storing and exchanging data. It stands for JavaScript Object Notation, but you don't need to know JavaScript to work with it. JSON is a lightweight, human-readable format that's easy to understand and generate. JSON files use a hierarchical structure, similar to XML, but with a simpler syntax. They consist of key-value pairs, arrays, and nested objects. This makes them great for organizing and structuring data. Here the datasets have been taken in a JSON file. Here 2 JSON files are prepared for training and testing purposes accordingly.

3.1.1. Training:

An intent in a dataset refers to the goal or purpose that a user has when interacting with a system. In the context of natural language processing and chatbots, intents are often used to classify user input. Here intents are divided into 3 parts i.e., tags, patterns, and responses. Tags show one crucial word that is included in all of the questions/queries. This word helps to organize the questions and make it easier for others to understand. Patterns are the anticipated questions and responses are the answers if those questions arise. It is stored in a JSON file and used later for training purposes Here 105 intents are taken for training.

3.1.2. Testing

The test samples provided at used to validate the model after training, ensuring that the chatbot can accurately identify the intents of new inputs that it has not seen during training. This helps in evaluating the generalization ability of the model. Here 21 intents are taken for testing purposes.

3.2. Preprocessing:

Tokenizing the input patterns into words. Removing common punctuation marks. Lemmatizing words to their base form. Creating a vocabulary of unique words and a list of unique intent tags. Generating training data as bag-of-words vectors and corresponding one-hot encoded output vectors. Shuffling and converting the training data to NumPy arrays for use in training the neural network. These steps are crucial for preparing the raw text data into a structured format that can be fed into a deep learning model for training.

3.2.1. Data Cleaning:

Each word is lemmatized to its base form and converted to lowercase. Words that are in the ignore_words list (punctuation marks) are removed. The list of words is sorted and duplicates are removed by converting it to a set and back to a list.

3.2.2. NLP:

In Natural Language Processing (NLP), the different stages of processing human language into computationally understandable structures involve several layers of analysis. Here's an explanation of each stage, focusing on their general meanings and applications within NLP:

- Lexical Analysis: Lexical analysis is the process of breaking down a sentence or text into its smallest elements or tokens. It is often called tokenization, where the text is split into words, phrases, symbols, or other meaningful units. This is the first step in processing natural language, allowing other stages to work with structured input. In NLTK (Natural Language Toolkit), you might use functions like `word_tokenize` or ‘sent_tokenize’ to perform lexical analysis.

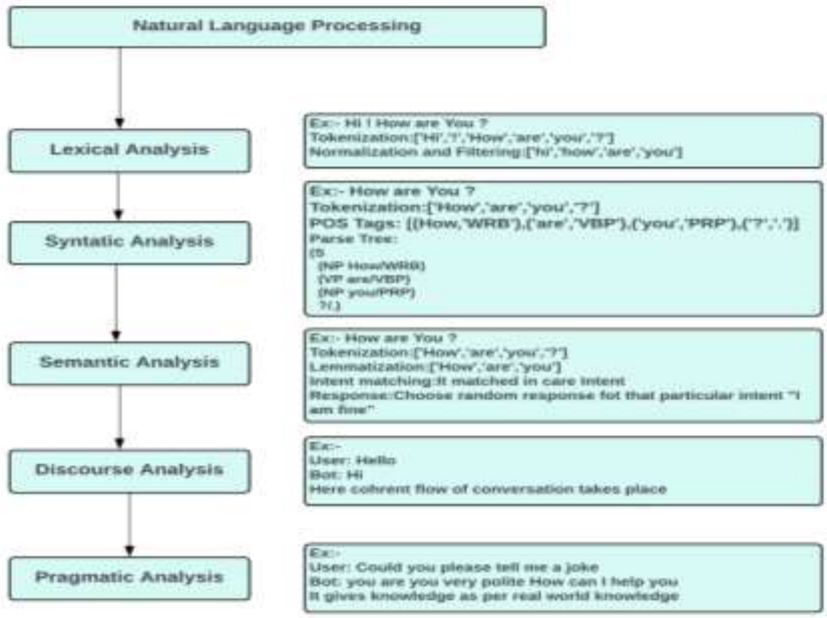
- Syntactic Analysis: Syntactic analysis, or parsing, examines the structure of sentences to determine the grammatical relationships between words. This stage helps to understand the syntax or grammar of the text. The output often involves a parse tree representing the grammatical structure.

- Semantic Analysis: Semantic analysis focuses on the meaning of words, phrases, and sentences. It aims to derive a representation that captures the intended meaning and contextual information. This can involve resolving ambiguity, understanding word sense, and extracting relationships. In NLTK, semantic analysis can involve word sense disambiguation (WSD) with tools like ‘lesk’ or working with semantic representations, such as semantic networks or ontologies.

- Discourse Analysis: Discourse analysis looks at the structure of larger units of text or conversation, focusing on how sentences and phrases connect to form coherent narratives or arguments. It explores relationships between sentences and the context within a larger conversation or document. In NLTK, you can work with discourse-related tasks, such as coreference resolution, which identifies when different words or phrases refer to the same entity.

- Pragmatic Analysis: Pragmatic analysis involves understanding the context and intent behind language use. It considers factors like speaker intention, social context, and the implications of what is said. Pragmatics goes beyond the literal meanings to capture subtleties like sarcasm, politeness, or humor. Pragmatic analysis in NLTK may not be as directly implemented as other analyses but often involves a combination of semantic and discourse analysis to infer context and intent.

These stages together enable a comprehensive understanding of natural language, with each step contributing to deeper insights into the text or speech being analyzed. Below Fig.1. derives the NLP working briefly. Lexical analysis is Handled by tokenization (nltk,word_tokenize) Semantic analysis is Partially handled by lemmatization (WordNetLemmatizer) and training the neural network to understand patterns and intents. Syntactic analysis is Not explicitly handled, though the bag-of-words model captures some syntactic information. Discourse analysis is handled, and each user input is treated independently. Pragmatic analysis Not explicitly handled relies on predefined intents.



Ex:- Hi ! How are You ?
Tokenization:['Hi','!','How','are','your','?']
Normalization and Filtering:['hi','how','are','you']

Ex:- How are You ?
Tokenization:['How','are','you','?']
POS Tags: [(How,'WRB'),('are','VBP'),('you','PRP'),('?','.')]
Parse Tree:
(S
 (NP How/WRB)
 (VP are/VBP)
 (NP you/PRP)
 ?/.)

Ex:- How are You ?
Tokenization:['How','are','you','?']
Lemmatization:['How','are','you']
Intent matching:It matched in care Intent
Response:Choose random response fot that particular intent "I am fine"

Ex:-
User: Hello
Bot: Hi
Here cohrent flow of conversation takes place

Ex:-
User: Could you please tell me a joke
Bot: you are you very polite How can I help you
It gives knowledge as per real world knowledge

Fig.1. NLP Steps used in the proposed system

3.3. Classification Model:

The input layer is the entry point for the data fed into the neural network. The first hidden layer consists of 128 neurons. Each neuron in this layer receives input from every neuron in the input layer. The second hidden layer consists of 64 neurons. Each neuron in this layer receives input from every neuron in the first hidden layer. The output layer consists of 105 neurons. Each neuron in this layer receives input from every neuron in the second hidden layer (fully connected). The number of neurons in the output layer typically corresponds to the number of output classes in a classification problem or the number of outputs in a regression problem. In the below Fig.2. defines the chatbot model architecture.

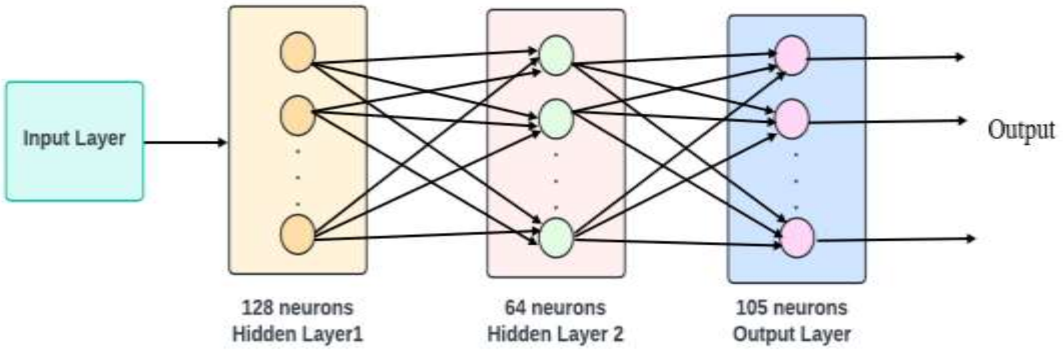


Fig.2. Architecture of Chatbot model

3.4. Chatbot:

3.4.1. Block Diagram:

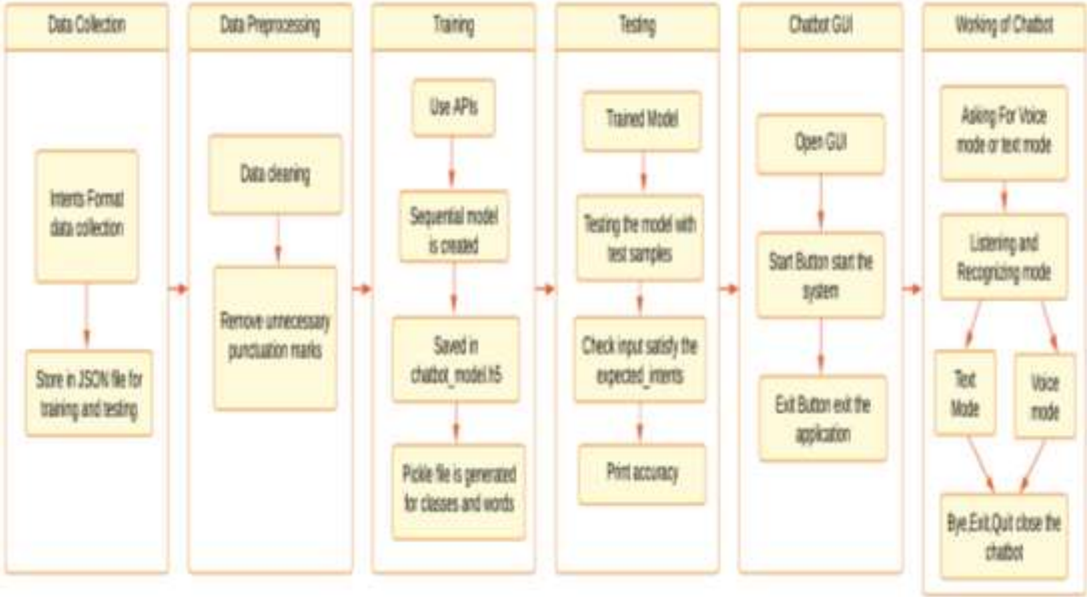

Data Collection
Intents Format data collection
Store in JSON file for training and testing
Data Preprocessing
Data cleaning
Remove unnecessary punctuation marks
Training
Use APIs
Sequential model is created
Saved in chatbot_model.h5
Pickle file is generated for classes and words
Testing
Trained Model
Testing the model with test samples
Check input satisfy the expected_intents
Print accuracy
Chatbot GUI
Open GUI
Start Button start the system
Exit Button exit the application
Working of Chatbot
Asking For Voice mode or text mode
Listening and Recognizing mode
Text Mode
Voice mode
Bye,Exit,Quit close the chatbot

Fig.3. Block Diagram of Anglips chatbot

Above Fig.3. displays the whole process of building a chatbot and implementation in real-time in both speech as well as text-based. Initially, data is collected in intents format and it is stored in a JSON file for both training and testing purposes then data-cleaning takes place by removing the punctual notation for a better understanding of the chatbot. After that process training was done by using the intent dataset and creating a model. Which is stored in the chatbot-model.h5 and generates some pickle files for words and classes. The next step is testing the model with the help of another JSON file which has some intents then accuracy is printed. Then fully user-friendly platform was developed by using the Qt-designer tool. Next is implemented with the help of PyQt5 in implementation. Chatbot uses a Speech Recognition API that recognizes voice and converts it into machine-understandable language as shown in Fig.4. shows how Speech recognition works. Chatbot is developed for both voice and text-based which makes it more efficient.

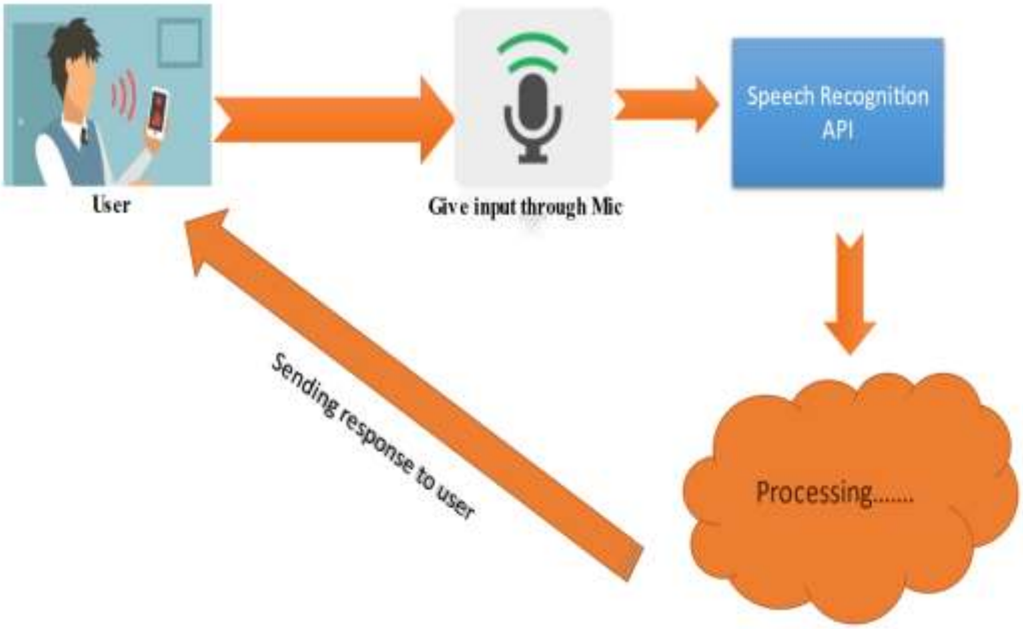


Fig.4. Working of Speech recognition API

3.4.2. Flowchart:

Anglips is a both voice and text-based chatbot. It's designed to provide an intuitive and engaging conversational experience for users. It will start by opening an intuitively designed graphical user interface (GUI) crafted with the Qt Designer tool. This interface will be implemented using PyQt5, a powerful framework for creating native applications with Python. When the system first launches, it will present the user with a Start button. Upon clicking the Start button, the chatbot will prompt the user to choose between text and voice-based interaction. This choice will determine the mode in which the chatbot will engage with the user for the duration of the conversation. After the initial setup, the chatbot will enter listening mode the chatbot will actively check the user's responses and go to recognizing mode. If the response is voice-based, the chatbot will continue the conversation in voice mode. If the response is text-based, the chatbot will switch to text mode and proceed with the conversation. The conversation will continue until the user issues the command "goodbye" or "exit" statement. At that point, the chatbot will gracefully terminate the conversation and recursively it goes to Listening and Recognizing mode. To end the session the user can simply click the Exit button, which will promptly close the application. Anglips is designed to provide a seamless and interactive conversational experience, blending the best of voice and text-based interaction which is graphically represented in flowchart Fig.5.

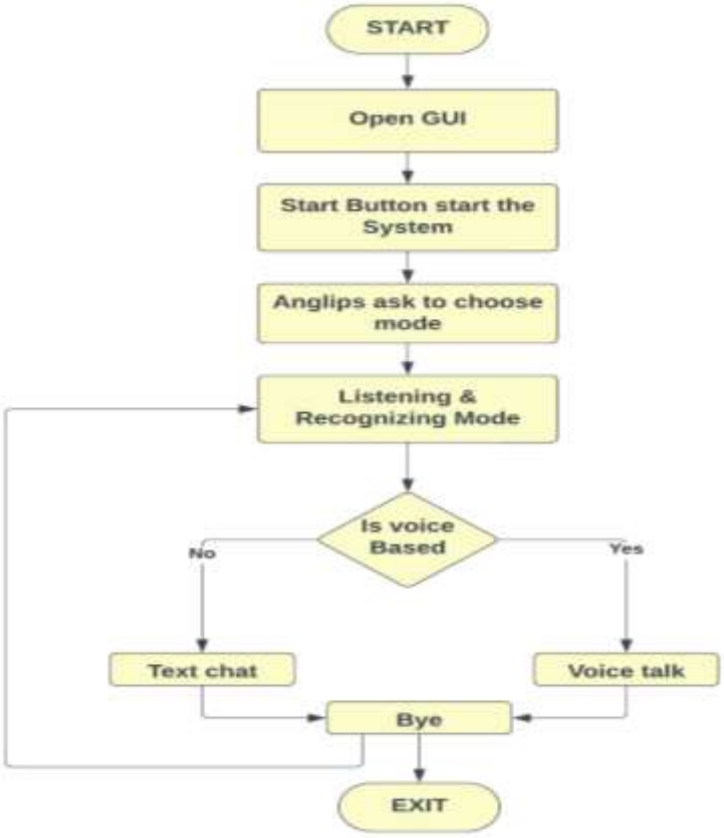

START
Open GUI
Start Button start the System
Anglips ask to choose mode
Listening & Recognizing Mode
Is voice Based
No
Yes
Text chat
Voice talk
Bye
EXIT

Fig.5. Flowchart of Anglips chatbot

4. Results and Discussion

4.1. Training & Testing

The dataset is processed using different APIs such as nltk, TensorFlow, and keras. which employ deep learning, neural networks, and NLP. It undergoes intensive training for 600 epochs, during which its performance is meticulously evaluated. This rigorous training ultimately leads to the development of a well-evaluated model. A sequential model is created. After training, the model is saved to a file (‘chatbot_model.h5’) for later use. Supporting data (‘words’, ‘classes’) is saved using ‘pickle’ for easy loading in future chatbot applications.

Trained Keras model (`chatbot_model.h5`) and supporting data ('words.pkl', 'classes.pkl') are used for testing the intents. A JSON file containing test samples. The JSON file is expected to contain a list of test samples, with each sample containing a "pattern" (input sentence) and an "expected_intent" (the correct intent for that sentence). It is intended to assess how well the chatbot model predicts the correct intent based on test samples. It provides a straightforward way to evaluate the chatbot's accuracy and can help guide further model training and improvements. Here the model is showing 95.24% accuracy. In below Fig.6. shows the accuracy of the training and testing, it distinguishes how well the model classifies the intents.

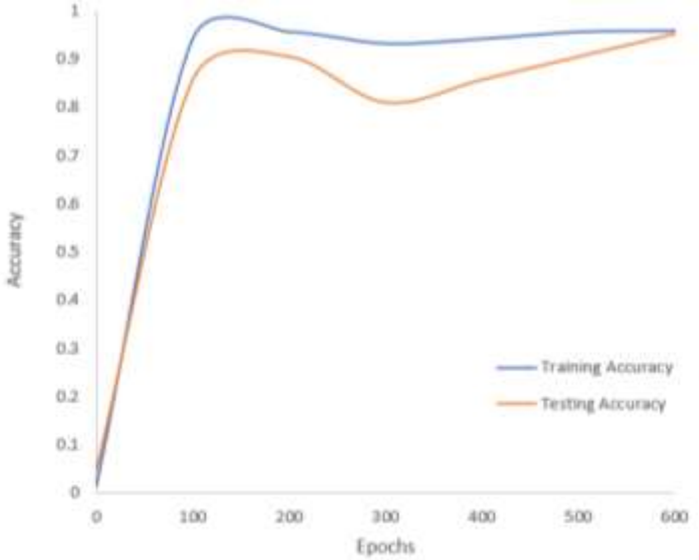


Fig.6. Accuracy graph

In below Fig.7. show the loss refers to the error between the predicted outputs of the model and the actual target values. It is a measure of how well the model's predictions match the actual data. Lower loss values indicate better model performance.

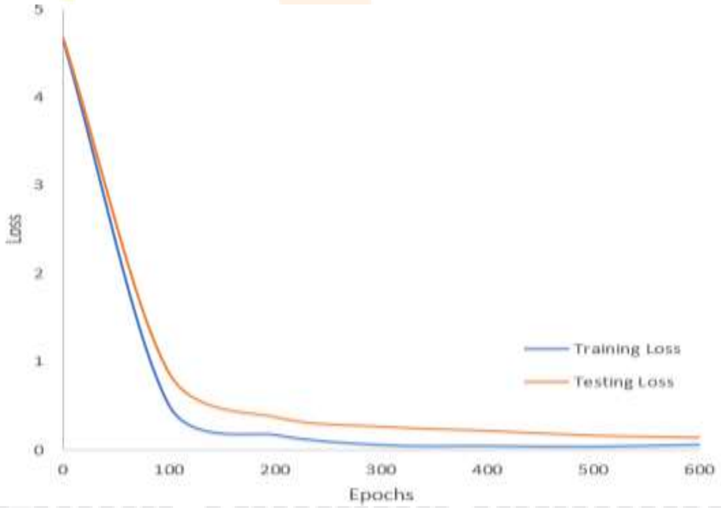


Fig.8. Loss graph

4.2. Comparison with Existing system:

Here the proposed Anglips chatbot has both voice and text-based modes. It has given a User-friendly, easy-to-use GUI that has been developed by Qt designer Tool and implemented in the system by using PyQt5 in Python.Below Table 1. shows the comparison between the existing chatbots.

Table 1. Comparison Against the Existing chatbots

Chatbot System	Accuracy	Modes
Jolity chatbot [1]	90%	Text-based
Amharic chatbot [9]	91.55%	Text-Based
Proposed Anglips chatbot	95.24%	Both Voice and Text-Based

4.3. Snapshots of the chatbot's working:

The system has been constructed using Python along with the Qt Designer tool for crafting the graphical user interface (GUI). Fig.8. illustrates the initial appearance of the GUI, featuring prominent elements such as the START and EXIT buttons. An innovative approach involves integrating voice interaction through a specialized API known as the Speech Recognition API. This API facilitates the system in soliciting user input audibly, enhancing accessibility and user experience.

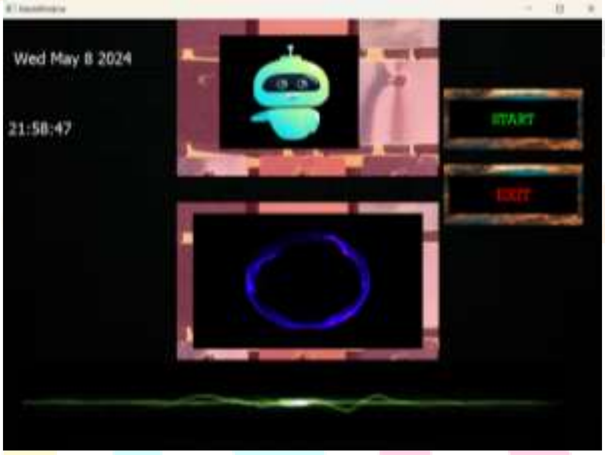
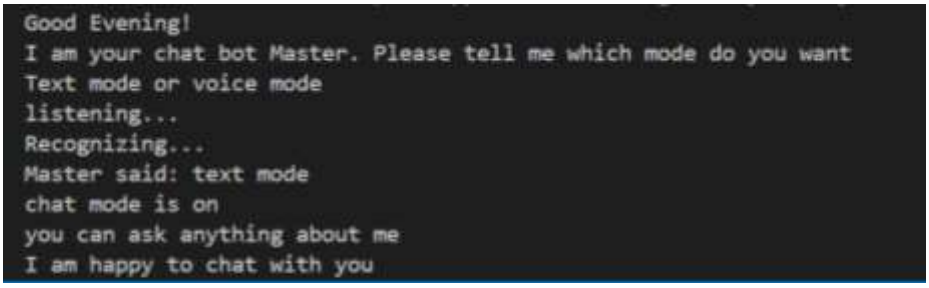


Fig.8. Chatbot GUI

The rise of speech recognition APIs represents a significant milestone in the history of human-computer interaction. By making it possible to interact with technology through natural speech, these APIs are breaking down barriers and making technology more accessible and inclusive. As developers continue to innovate with these powerful tools, we can look forward to a future where speech recognition plays an integral role in shaping the way we interact with the digital world.

In Fig.9. demonstrates the chatbot asking the user to choose text mode or voice mode.

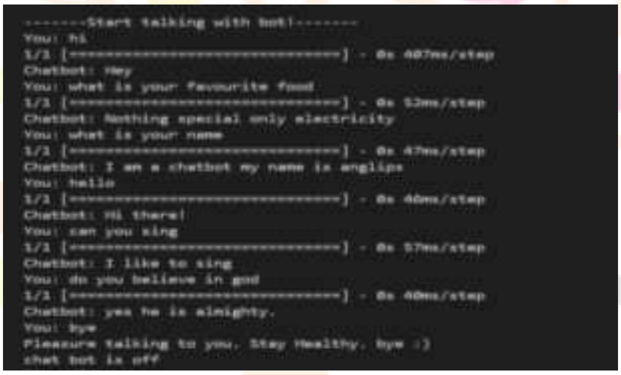


```
Good Evening!
I am your chat bot Master. Please tell me which mode do you want
Text mode or voice mode
listening...
Recognizing...
Master said: text mode
chat mode is on
you can ask anything about me
I am happy to chat with you
```

Fig.9. Choose mode

Fig.10. demonstrates an efficient text-based conversation between a chatbot and a user. The diagram likely illustrates a clear, responsive interaction in which the chatbot promptly and accurately addresses the user's inquiries. This exchange might involve a variety of features that contribute to a smooth conversational flow, such as natural language understanding, context awareness, and rapid response times.

Additionally, Fig.10. might highlight the chatbot's ability to manage different types of user inputs, providing coherent responses even when users ask complex questions or change topics. The chatbot's efficiency could also be attributed to its capacity for task automation, allowing users to complete specific actions, like booking appointments or retrieving information, without needing additional assistance.

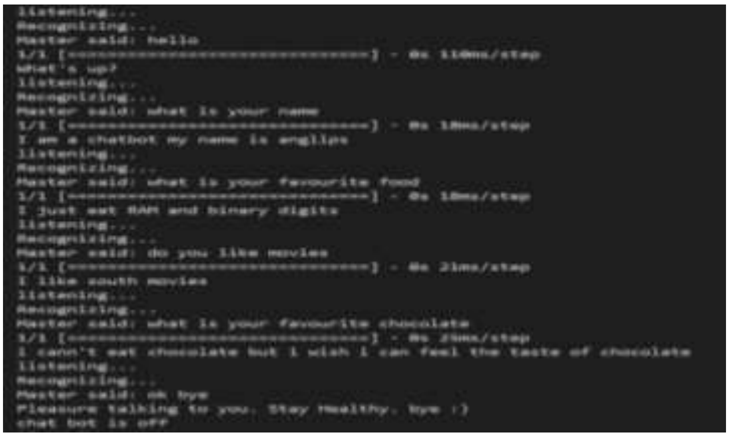


```
--------Start talking with bot!--------
You: hi
1/1 [==============================] - 0s 407ms/step
Chatbot: Hey
You: what is your favourite food
1/1 [==============================] - 0s 52ms/step
Chatbot: Nothing special only electricity
You: what is your name
1/1 [==============================] - 0s 47ms/step
Chatbot: I am a chatbot my name is anglips
You: hello
1/1 [==============================] - 0s 40ms/step
Chatbot: Hi there!
You: can you sing
1/1 [==============================] - 0s 57ms/step
Chatbot: I like to sing
You: do you believe in god
1/1 [==============================] - 0s 40ms/step
Chatbot: yes he is almighty.
You: bye
Pleasure talking to you. Stay Healthy. bye :)
chat bot is off
```

Fig.10. Talking with the chatbot in Text mode

In Fig.11. the focus is on voice-based communication, illustrating how a chatbot interacts with users through spoken dialogue. This type of interaction involves several key elements that make it both functional and user-friendly. The core of voice-based interaction is speech recognition. The chatbot must be able to accurately interpret what the user is saying. This requires sophisticated technology capable of recognizing various accents, speech patterns, and sometimes even background noise. The more accurately the chatbot can transcribe spoken words, the smoother the conversation will be. Once speech is recognized, the next step is natural language understanding (NLU). This involves determining the intent behind the user's words and extracting any relevant information.

This step is crucial for the chatbot to provide meaningful responses and take appropriate action based on the user's request. Fig.11. showcases a voice-based chatbot that can understand, respond, and interact with users effectively, offering a seamless and engaging conversational experience.



```
listening...
Recognizing...
Master said: hello
1/1 [==============================] - 0s 110ms/step
What's up?
listening...
Recognizing...
Master said: what is your name
1/1 [==============================] - 0s 18ms/step
I am a chatbot my name is angllps
listening...
Recognizing...
Master said: what is your favourite food
1/1 [==============================] - 0s 18ms/step
I just eat RAM and binary digits
listening...
Recognizing...
Master said: do you like movies
1/1 [==============================] - 0s 21ms/step
I like south movies
listening...
Recognizing...
Master said: what is your favourite chocolate
1/1 [==============================] - 0s 29ms/step
I cann't eat chocolate but i wish i can feel the taste of chocolate
listening...
Recognizing...
Master said: ok bye
Pleasure talking to you. Stay Healthy. bye :)
chat bot is off
```

Fig.11. Talking with the chatbot in Voice mode

5. Conclusion and Future work

Chatbots have revolutionized the way we interact with technology, providing instant, personalized responses across various platforms. Their applications range from customer service to healthcare, education, and entertainment. By leveraging advancements in natural language processing (NLP) and machine learning, chatbots can understand and respond to user inputs with increasing accuracy and relevance. Chatbots are not only used for technical support FAQs, and Help desks but also help in chatting and sharing personal feelings which makes people more relaxed and reduces extra stress. The development of chatbots involves meticulous training and validation processes. Accuracy and loss metrics are crucial for assessing their performance. Training accuracy and loss indicate how well the model learns from the data, while validation accuracy and loss reveal its ability to generalize to new, unseen inputs. Striking a balance between these metrics is essential to prevent overfitting, where the model performs well on training data but poorly on validation data, or underfitting, where the model fails to capture underlying patterns in the data. Despite significant progress, current chatbot technology faces several limitations. These include handling ambiguous or complex queries, maintaining context in long conversations, and providing emotionally intelligent responses. Additionally, chatbots can struggle with understanding diverse linguistic nuances and accents, which can affect user experience. Developing chatbots that can recognize and respond to human emotions will create more empathetic and engaging user experiences. Incorporating sentiment analysis and affective computing will be crucial in this regard. Implementing more sophisticated user profiling and personalization techniques will allow chatbots to offer tailored responses and recommendations, enhancing user satisfaction.

Reference

[1] Deepika, K., Tilekya, V.,Mamatha, J. & Subetha, T (2020). Jollity Chatbot: A Contextual AI Assistant. (ICSSIT). IEEE.

[2] Dahiya, M. (2017). A Tool of Conversation: Chatbot.International Journal , 5(5), 158. E-ISSN: 2347-2693.

[3] Kasthuri, E., & Balaji, S. (2023). Natural Language Processing and Deep Learning Chatbot Using Long Short-Term Memory Algorithm. *Material Today: Proceedings* 81, 690-693.

[4] Bhavya, M.R., Prajwal, Y.P., Roushan, M., Sidiq, M.A.P., & Hemmige, S.K. (2023). Intelligent Chatbot. *International Research Journal of Modernization in Engineering, Technology and Science*, 5(5), 6552. e-ISSN: 2582-5208.

[5] Lalwani, T., Bhalotia, S., Pal, A., Bisen, S., & Rathod, V. (2018). Implementation of a Chatbot System using AI and NLP. International Journal of Innovative Research in Computer Science & Technology (IJIRCST), 6(3), 26. DOI: 10.21276/ijircst.2018.6.3.2. ISSN: 2347-5552.

[6] Sonawane, M., Kasar, S., Pednekar, S., & Suvare, S. (2021). Chatbot for College Information System. Journal of Emerging Technologies and Innovative Research (JETIR), 8(5), b898. ISSN: 2349-5162.

[7] Bala, K., Kumar, M., Hulawale, S., & Pandita, S. (2017). Chat-Bot for College Management System Using A.I. *International Research Journal of Engineering and Technology (IRJET)*, 4(11), 2030. e-ISSN: 2395-0056, p-ISSN: 2395-0072.

[8] Dhyani, M., & Kumar, R. (2021). An Intelligent Chatbot Using Deep Learning with Bidirectional RNN and Attention Model. *Material Today: Proceedings* 34, 817-824.

[9] Ybrah, G., & Welay, S. (2024). Deep Learning-based Amharic Chatbot for FAQs in Universities.

[10] Ranoliya, B.R., Raghuwanshi, N., & Singh, S. (2018). Chatbot for University Related FAQs.