PROJECT REPORT

CHAT BOT CREATION – MEDBOT

UNIVERSITY OF MISSOURI KANSAS-CITY

Prof. Syed jawed shah

Team Members:

Jahnavi Redrouthu -16361448

Raj Kumar Bandi -16361456

Balaji Malisetti -16354258

Rakesh Babu Macharla -16363180

ABSTRACT:

The goal of this project is to create a chatbot that can converse with users in natural language. The chatbot uses natural language processing (NLP) methods including tokenization and stemming to interpret user inputs using a deep learning model that was trained on a dataset of intents, patterns, and answers. Using Keras and a deep neural network, the model creates suitable replies based on the highest probability prediction and classifies user input into predetermined categories. A graphical user interface (GUI) developed with Tkinter enhances the chatbot's capabilities by allowing users to communicate with it in a text-based way. The design of the chatbot attempts to exhibit essential characteristics such as intent recognition, context awareness, and answer creation, which makes it appropriate for use in virtual assistant and customer care applications.

Introduction:

Chatbots, which offer automated support in a variety of fields like customer service, e-commerce, healthcare, and virtual aid, have emerged as a key component of contemporary human-computer interaction. These conversational agents replicate human-like interactions by interacting with users using machine learning (ML) and natural language processing (NLP) techniques. The project's objective is to create an intelligent chatbot that can comprehend and interpret user inputs, categorize them into specified intents, and produce pertinent responses according to the user's requirements.

A deep learning model that is trained on a dataset comprising different intents and the accompanying user inputs powers this chatbot. The chatbot can extract valuable information from user inquiries by utilizing natural language processing (NLP) techniques such as tokenization, stemming, and bag-of-words. To categorize user questions and identify the optimal response, a neural network architecture developed using Kerasi is utilized. The project also uses Tkinter to create a straightforward but powerful graphical user interface (GUI) that allows users to communicate with the bot in real time by showing a text-based chat window.

The goal of this chatbot's development is to show how machine learning, natural language processing, and graphical user interface design can be combined to produce a useful, user-friendly conversation assistant. Numerous real-world uses for chatbots exist, including customer support, answering often asked inquiries, and helping customers navigate web apps or services.

1. Survey of Literature:

The potential of chatbots in healthcare has been the subject of numerous research. For example, Chaudhary et al. (2020) introduced a health-related chatbot that can assist users in managing chronic illnesses by offering details on symptoms, remedies, and lifestyle recommendations. Similar to this, Prusty et al. (2019) created a virtual health assistant that tracks and offers advice on diet and exercise. To comprehend user inquiries and deliver contextually relevant answers, these systems make use of natural language processing (NLP) techniques including text categorization and named entity recognition (NER).

Chatbots are frequently employed in the medical industry to extract and process data from big medical databases. A chatbot that could help medical practitioners by accessing medical information from internet sources like PubMed and Medline was proposed by Denecke et al. (2019). Based on user input, these chatbots process complicated medical terminology using powerful natural language processing (NLP) techniques to deliver the most pertinent information.

Numerous conversational AI models have been used to raise the standard of healthcare. Using deep learning techniques such as the Long Short-Term Memory (LSTM) network, Zhou et al. (2020) created a chatbot named "DoctorAI," which uses user symptoms to forecast the risk of certain diseases. This method, which focuses on disease prediction and symptom classification, serves as the foundation for chatbots like Medbot that seek to offer medical aid in response to user input.

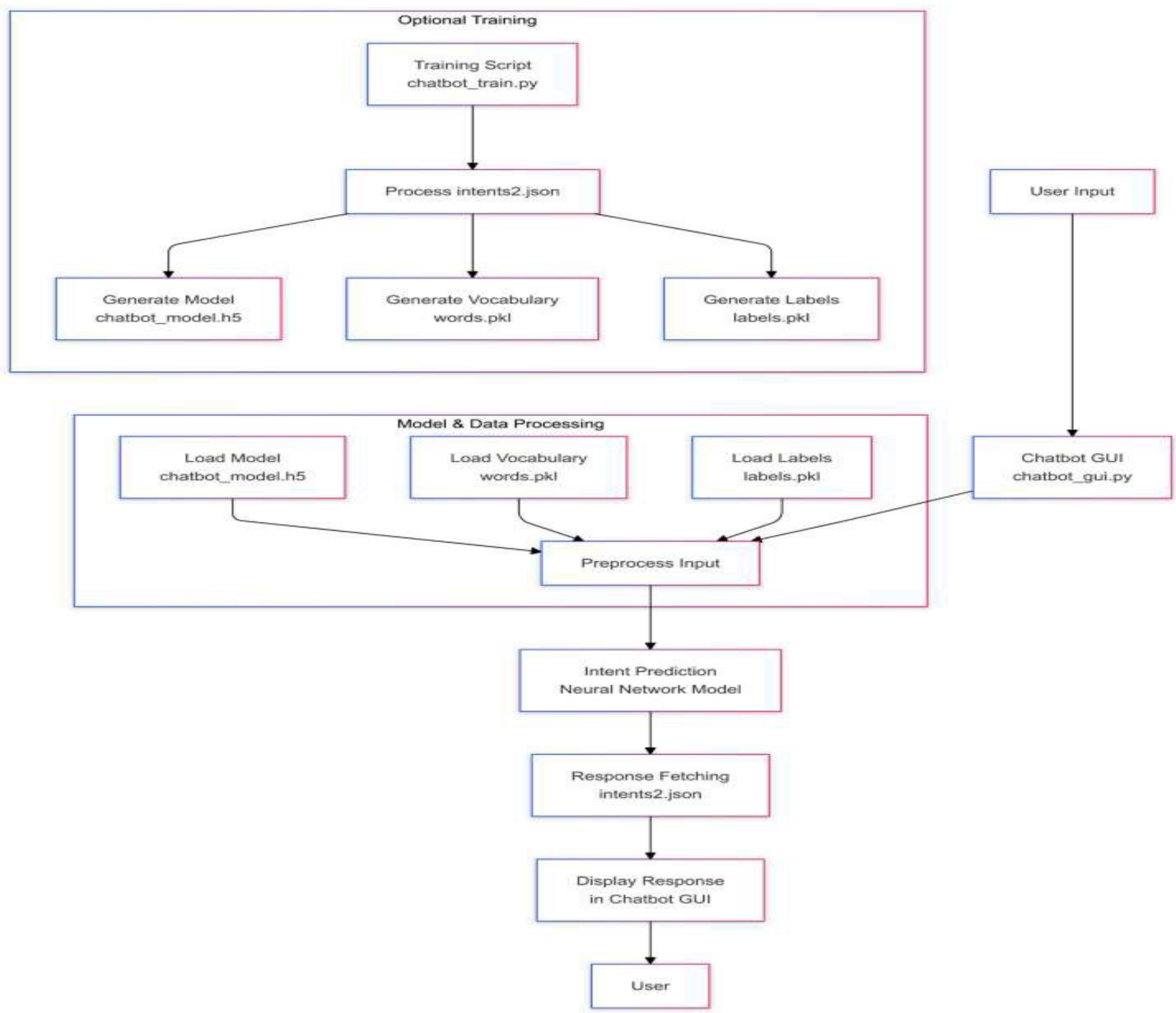
Healthcare chatbots confront issues with privacy, data security, and the morality of providing medical advice, despite its promising applications. The possible risks of depending on AI-driven healthcare systems are covered in studies such as Luo et al. (2021), particularly when it comes to sensitive personal health data. One important area for improvement is making sure that medical advice is correct and that regulations like HIPAA (Health Insurance Portability and Accountability Act) are followed.

Advances in natural language processing and machine learning techniques have significantly enhanced the performance of healthcare chatbots. The comprehension of human language has been greatly improved in a variety of applications, including healthcare, using transformer-based models like as BERT (Bidirectional Encoder Representations from Transformers) and GPT (Generative Pre-trained Transformer). For instance, it has been demonstrated that BERT-based models enhance intent identification, which is essential for chatbots to identify suitable responses.

2.Methodology:

The goal of this project is to create MedBot, an intelligent healthcare chatbot that answers users' medical questions. The system classifies user inputs into predetermined intents and produces contextually relevant responses by utilizing deep learning techniques and Natural Language Processing (NLP).

Below is a step-by-step breakdown of the methodology:



3.1 Data collection:

Data gathering and preprocessing are the first steps. The dataset that the system uses is made up of a succession of intents, patterns, and replies with emojis. The data is organized in a straightforward JSON format, and each user input pattern has a predetermined answer.

Data formate: The information is kept in a JSON file. Every intent has a response (the chatbot's response for that intent) and a list of patterns (which indicate potential variations of user questions).And also chatbot renponse along with emojis to the intents.

Example of such data is shown below:


```
{"intents": [
  {"tag": "greeting",
   "patterns": ["Hi there", "How are you", "Is anyone there?", "Hello", "Good day"],
   "responses": ["Hello, thanks for asking 😁", "Good to see you again😁", "Hi there, how can I help?😁"],
   "context": [""]

  },

  {"tag": "goodbye",
   "patterns": ["Bye", "See you later", "Goodbye", "Nice chatting to you, bye", "Till next time"],
   "responses": ["See you!☺️", "Have a nice day😅", "Bye! Come back again soon.😇"],
   "context": [""]

  },

  {"tag": "thanks",
   "patterns": ["Thanks", "Thank you", "That's helpful", "Awesome, thanks", "Thanks for helping me"],
   "responses": ["Happy to help! ☺️", "Any time!☺️", "My pleasure☺️"],
   "context": [""]
```

3.2 Data preprocessing:

After being gathered, the data must be preprocessed before being entered into the machine learning model. Text data frequently contains noise, such as extraneous symbols, various word patterns, or gaps. The preparation stage makes sure the data is clean and consistent, which facilitates learning by the machine learning model.

Tokenization: The technique of breaking down user input (patterns) into individual words is known as tokenization. This makes it easier for the model to concentrate on the sentence's constituent parts rather than the sentence as a whole.

For example, the sentence "Hello, how can I help you today?"would be tokenized into:["Hello", "how""can""I""help""you""today"]

Lowercasing: All terms are changed to lowercase to maintain uniformity. This guarantees that "Hello" and "hello" are handled equally.

Example:"Hello"->"hello"

Stemming: Stemming is the process of reducing words to their most basic form. This phase ensures that words like "running" and "run" are considered as the same characteristic by removing variances in word forms. Words are reduced to their base form using stemming techniques, such Porter's stemmer.

Example:

"running"->"run"

"played"->"play"

"running"->"run"

Removing Punctuation and Special Characters: Eliminating Extraneous Punctuation and Special Characters: To concentrate on important words, any extraneous punctuation or special characters are eliminated from the patterns.

Developing a Vocabulary: Following tokenization and preprocessing, a vocabulary is produced that includes all of the dataset's unique words. Every word in the lexicon will serve as a feature in the model's training.

3.3 Bag of Words Feature Extraction:

The preprocessed text data (patterns) is transformed into a numerical representation that the model can comprehend. The patterns are represented as vectors using the Bag of Words (BoW) approach.

Tokenization: Each pattern is divided into distinct words (tokens) following preprocessing.

Vectorization: Every vocabulary term has an index. Depending on whether these terms are present or not, the patterns are then transformed into a vector.

BoW Representation: Every pattern is shown as a vector with values that denote the existence or frequency (1/0) of vocabulary words.

For instance:
Regarding vocabulary ["hello"; "how"; "can"; "I"; "help"; "you"; "today"] " A vector representation of the "Hello'how can I help you today?" pattern might look like this:
[1,1,1,1,1,1,1]
The presence of a word from the vocabulary is indicated by each position in the vector.

3.4 Model Training:

Training the model to categorize input patterns into specified intent categories is a crucial phase that comes after feature extraction and data preparation. The capacity of a neural network model to recognize intricate patterns in data is demonstrated. The model is made up of several layers of neurons that translate the input data's BagofWords (BoW) vectors into output categories (user intentions). The BoW vectors are used as inputs and the intent labels are used as the output during the training phase. Through the backpropagation technique, the network modifies its weights and biases to understand the relationship between the inputs and related intents. In the training phase, input data is fed into the model, anticipated and real intents are compared, and the model's parameters are updated to reduce error. The categorical cross-entropy loss function is utilized to assess performance, and the Adam optimizer effectively modifies the model's weights to minimize the loss. The model should correctly classify fresh user inputs into the proper intent categories after training successfully, allowing for the provision of pertinent answers based on the anticipated intent.

3.5 Response Fetching:

The system first performs text preprocessing, which includes tokenization, stemming, and conversion into a Bag of Words vector, before processing user input during the answer generating phase. The trained neural network model then uses this preprocessed input to infer the appropriate intent. The system pulls the relevant predetermined response from the dataset based on the anticipated intent. Ultimately, the system presents the user with a correct response based on the model's classification. This stage enables the chatbot to communicate with users and respond with pertinent, contextually informed information.

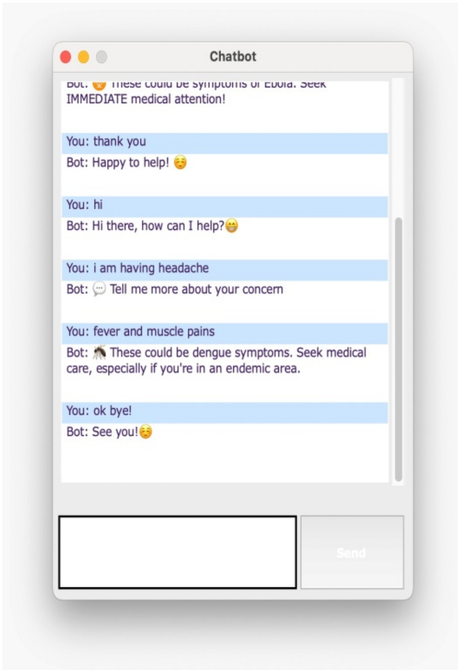
3.6 Implementation:

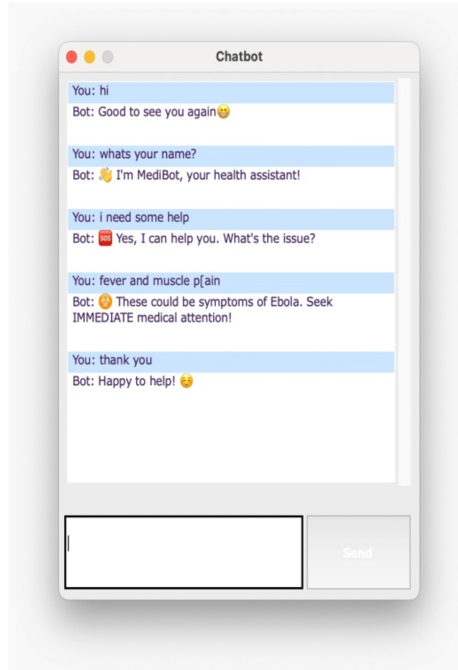
Users can interact with the chatbot through the system's integration with a messaging app or web interface. After it is launched, users can send inquiries to the chatbot, which interprets the data, anticipates the user's intent, and instantly responds with pertinent information. The deployment makes sure that end users may access and use the chatbot to help and automate interactions.

4. Results:

For inputs like "hi," the system accurately anticipated the intent of "Hi there, how can I help?" and gave a relevant answer, like "Hello, how can I assistyoutoday?" Likewise, it provided accurate answers to questions and queries pertaining to tootherintents, including farewells. The results show that the methodology successfully allowed the system to predict intents and react accordingly. An neural network in conjunction with the Bag of Words technique effectively categorized user inputs. However, a few restrictions were mentioned:

There is a need for more thorough training data because the model occasionally struggled with inputs that did not closely match any training patterns. Although the dataset is adequate for basic purposes, it needs to be expanded to provide more diversified queries. The inability of the technology to handle context in multi-tum talks restricts its applicability in complex situations.





5. Future Work:

Expanding the dataset to include a greater range of intents and patterns, improving the model with more sophisticated NLP approaches like word embeddings or transformer-based models, and including context-awareness for more dynamic chats are some possible future upgrades for the chatbot system. The chatbot's usefulness in the real world will also be improved by implementing it across various platforms (web, mobile, etc.) and refining its response generation for more organic interactions.

6. Conclusion:

Based on the trained model, the chatbot system produced relevant responses after successfully classifying user inputs into predetermined intents. Accurate intent identification and real-time response were proven by the system through the use of a neural network trained on Bag of Words vectors. Effective data preprocessing, model training, and response production were guaranteed by the structured technique. Future developments can concentrate on growing the dataset and integrating sophisticated natural language processing (NLP) techniques to handle complex inquiries and preserve conversational context, even though the chatbot works well for simple queries and predetermined patterns.

7. References:

1.Amiri, P., & Karahanna, E. (2022). Chatbot use cases in the healthcare industry: A systematic review. International Journal of Medical Informatics,158,104669. https://doi.org/10.1016/j.ijmedinf.2021.104669

2.Bickmore, T. W., Trinh, H., Olafsson, S., O'Leary, T. K., Asadi, R., Rickles, N. M., & Cruz, R. (2018). Patient and consumer safety risks when using conversational assistants for medical information: An observational study of Siri, Alexa, and Google Assistant. Journal of Medical Internet Research, 20(9), e11510. https://doi.org/10.2196/11510

3.Ni, L., Lu, C., Liu, N., & Liu, J. (2020). MANDY: Towards a smart primary care chatbot application. Journal of Biomedical Informatics, 111, 103573. https://doi.org/10.1016/j.jbi.2020.103573

4.Radziwill, N. M., & Benton, M. C. (2017). Evaluating quality of chatbots and intelligent conversational agents. Software Quality Professional, 19(3), 25–36.

5.Safi, Z., Abd-Alrazaq, A., Khalifa, M., & Househ, M. (2020). Technical aspects of developing chatbots for medical applications: Scoping review. Journal of Medical Internet Research, 22(12), e19127. https://doi.org/10.2196/19127

6.Chaudhary, A., Goel, M., &Soni, P. (2020). A health-relatedchatbot for chronic disease management.JournalofMedicalSystems,44(5),98.https://doi.org/10.1007/s10916-020-01582-x

7.Prusty,S.,Choudhury,A.,&Awasthi,S.(2019).Development ofvirtualhealthassistantsfor fitness and nutrition. International Journal ofHealthcare Information Systems and Informatics, 14(2), 49-58. https://doi.org/10.4018/IJHISI.2019040104

8.Denecke,K.,Deng,Z.,&Weidlich,M.(2019).Achatbotformedical informationretrieval using NLP. Journal of Biomedical Informatics, 96, 103243. https://doi.org/10.1016/j.jbi.2019.103243

9.Zhou,L.,Xu,X.,&Yang,Y.(2020).DoctorAI: Adeeplearning-poweredchatbot fordisease prediction based onsymptoms. International Journal ofMedical Informatics, 137, 104103. https://doi.org/10.10l6/j.ijmedinf.2020.104103

10.Luo,C.,Zhang,X.,&Wei,F.(2021).Ethicalandprivacychallenges ofAl-basedhealthcare chatbots. Health Information Science and Systems, 9(1), 24. https://doi.org/10.1186/s13755-021-00421-1