

A Chatbot for Medical Purpose using Deep Learning

Himanshu Gadge

Computer Engineering
All India Shri Shivaji Memorial
Society COE, PUNE-1.
Savitribai Phule University Pune,
Maharashtra, India

Vaibhav Tode

Computer Engineering
All India Shri Shivaji Memorial
Society COE, PUNE-1.
Savitribai Phule University Pune,
Maharashtra, India

Sudarshan Madane

Computer Engineering
All India Shri Shivaji Memorial
Society COE, PUNE-1.
Savitribai Phule University Pune,
Maharashtra, India

Prateek Kachare

Computer Engineering
All India Shri Shivaji Memorial Society COE, PUNE-1.
Savitribai Phule University,
Pune, Maharashtra, India

Prof. Anuradha Deokar

Computer Engineering
All India Shri Shivaji Memorial Society COE, PUNE-1.
Savitribai Phule University Pune,
Maharashtra, India

Abstract--The Chatbot is a software programs that is used to interact with clients using natural language Processing via text or text to speech format. Today in the present era, the major challenges that India as a country is facing is to cater good quality and affordable healthcare services to its growing population and at the same time, they are not cost efficient. Nowadays, it is becoming very difficult to provide healthcare facilities as we have seen in COVID-19 critical situations that the condition in India was getting worse because of lack of transportation, availability of doctors and hospitality. Sometimes it causes the people to postpone their treatment as well as there is an increment in death count. The aim of our Project is to design a Conversational AI Powered Chatbot for Medical Diagnostics using Deep Learning which mainly focuses on rural parts as well as poor and needy people of our country. Our System has the capability to understand the symptoms of the patient and communicates with Patient (End-user) through web-UI. Our system tries to solve their problem with the help of the symptoms provided by Patient itself and help them to give the correct antibiotics/ medicines and precautions. NLTK (Natural Language Toolkit) is a module/ program in python which can able to perform symbolic and statistical Natural Language Processing for English written in Programming. It is used to analyze the input in the form of speech and generate responses that are similar to humans.

Keywords: Chatbot, COVID-19, Deep Learning, Natural Language Toolkit (NLTK), Natural Language Processing (NLP), Intent, bag of words, Flask.

I. INTRODUCTION

The Chatbot is a software programs that is used to interact with clients using natural language Processing via text or text to speech format. Initially, chatbot created for interact with human for entertainment purpose only. Today in the present era, the major challenges that India as a country is facing is to cater good quality and affordable healthcare services to its growing population and at the same time, they are not cost efficient. This inaccessibility of healthcare facilities especially in rural areas and the difficulty in accessing means of transport and it causes patients to postpone their treatment, or option for medical facilities which is closer but at the same

time are not cost-efficient and well-matched to their medical needs. The efficient ways to provide timely medical care, access and quality of the treatment to the patient, the role of Medical Chatbot comes into play which connects patients with the chatbot and help them to give the correct antibiotics/ medicines and precautions.

The typical interaction b/w humans and chatbot using Natural Language Processing (NLP) as follows:

1. A human interact with chatbot
2. The chatbot captures the audio of the human
3. The chatbot converts the Audio to text.
4. The Chatbot Process the text's data.
5. Data/text to audio conversion takes place in the chatbot.
6. The chatbot responds to the human in audio format.

Deep Learning is a subset of machine learning in Artificial Intelligence concerned with algorithms capable of learning unsupervised from data which is unstructured and unlabeled. It uses a function of the brain called neural networks.

➤ Neural Network:

Neural network is a sequence of algorithms that try to build the relationships between the data and the human brain. Neural networks adapt the changing input so, the network generates the best possible output without needs to redesign again.

➤ Modules:

1. TensorFlow:

It is an end-to-end Open-Source platform for Machine Learning. It has a different tools, libraries and community resources.

2. Speech Recognition:

It is an automatic speech recognition (ASR). It has the power which enables a program to process human audio into a written format.

3. *Natural Language Toolkit (NLTK) Libraries:*

It contains libraries and programs for Natural language processing. It is the most powerful NLP libraries, which contains packages to make chatbot understand human audio and reply to it with an accurate response.

4. *Flask Module:*

Flask is a lightweight WSGI web application framework. It is built to get started quick and easy, with the capability to scale up to the complex applications.

5. *Keras:*

It is a deep learning API written in Python language, running on the top of the machine learning platform i.e., Tensor flow. It is used to create layers in Neural Network.

II. LITERATURE REVIEW

The chatbot is a software program that is used to interact with clients using natural language. Humans can replace by chatbots for monotonous jobs of answering queries and giving efficient responses. [2] Chatbots used for acquiring knowledge. It can be implemented on our mobiles and local personal systems and can access the internet. Chatbots communicates with clients in any particular domain with their query as input in general conversational statements [2].

[7] This literature review presents History, Technology and Existing Work of the Chat-Bots. Chatbots have good conversational abilities they are good at emulating a human conversation, and they learn through knowledge [7].

[7] Alan Turing wondered if a Computer program could talk to a group of people without realizing that their interlocuter was artificial, this question was named Turing test. The first chatbot with ELIZA name was constructed in 1966. ELIZA simulated a psychotherapist's operation, returning the user's sentences in the interrogative form. ELIZA uses pattern matching and a response selection scheme based on templates. In the history of chatbots ALICE was the first online chatbot inspired by ELIZA. ALICE was based on pattern-matching [7]. Difference between ALICE and ELIZA was that ALICE was developed using new language AIML created for this purpose.[7] PARRY is considered more advanced than ELIZA as it is supposed to have personality and better controlling structure [7].

Chats-Bots mentioned in this Literature survey is also based on medicinal purpose such as K-Bot. [3] There is a well-recognized need for a shift to proactive asthma care given the impact asthma has on overall healthcare costs. The demand for continuous monitoring of patient's adherence to the medication care plan, assessment of environmental triggers, and management of asthma can be challenging in traditional clinical settings and taxing on clinical professionals [3]. K-Bot is limited to only Asthma and cannot treat on anything else.

[8] Deep learning is a subset of machine learning in artificial intelligence that has networks capable of learning unsupervised from data that is unstructured or unlabeled. Also known as deep neural learning or deep neural

network. Recurrent Neural Network (RNN) are a type of Neural Network where the output from previous step is fed as input to the current step. ... It uses the same parameters for each input as it performs the same task on all the inputs or hidden layers to produce the output [8].

Natural language processing (NLP) deals with building computational algorithms to automatically analyses and represent human language. In this paper, survey on Neural Network in Speech Recognition is Referred from the research paper speech recognition using deep neural networks published in 2019. [5] Deep learning algorithms have been mostly used to enhance the capabilities of computers so that it understands what humans can do, which includes speech recognition [5]. The main focus of the review was to shed a light on paper that employ the Deep learning technique for implementation with best use of neural network.[5] Machine learning is defined as the field of study that provides computers with the ability to learn from input data without being explicitly programmed to do so. Automatic speech recognition is the capability of a machine or computer to recognize the content of words and phrases in an uttered language and transform them to a machine-understandable format. Automatic speaker recognition can be defined as the process of recognizing the unknown speaker on the basis of the information embedded in his/her speech signal using machine [5]. Disadvantages of Speech Recognition were that it was difficult to distinguish the language of the user, also it may face difficulty in accent recognition as well.

COVID-19 has taken control all over world, millions of people have suffered from Covid and many have resulted in death. There is a survey based on COVID-19 which was implemented using Deep Learning. [6] This survey explores how Deep Learning has battled the COVID-19 pandemic and provides direction for future research on COVID-19. We cover Deep Learning applications in NLP, Computer Vision, Life Science and Epidemiology [6].

A. PAST WORK

1. ELIZA (1966)

Description: ELIZA simulated a psychotherapist's operation, returning the user's sentences in the interrogative form Weizenbaum ELIZA uses pattern matching and a response selection scheme based on templates.

Limitations: Its ability to communicate was limited, its knowledge is limited, it cannot keep long conversations and cannot learn or discover context from the discussion.

2. PARRY (1972)

Description: PARRY is supposed to have a "personality" and a better controlling structure. It defines his responses based on a system of assumptions and "emotional responses" activated by the change of utterances.

Limitations: Chatbot with low capabilities concerning language understanding and the ability to express emotions. It also has a low speed of responding.

3. ALICE (1995)

Description: ALICE was based on pattern-matching. ALICE's Knowledge Base consisted of about 41,000 templates and related patterns.

Limitations: ALICE did not have intelligent features and could not generate human-like answers expressing emotions or attitudes.

B. EXISTING WORK

1. kBot: Knowledge enabled personalized Chatbot for Asthma self-Management

Authors: Dipesh Kadariya, Revathy Venkataramanan, Hong Yung Yip, Maninder Kalra, Krishnaprasad Thirunarayanan.

Year: 2019

Description: kBot is a Chatbot mainly used to help Asthma patients through meaningful conversation.

Limitations: ALICE did not have intelligent features and could not generate human-like answers expressing emotions or attitudes.

2. PathBot: An Intelligent Chatbot for Guiding Visitors Locating Venues

Authors: Katlego Mabunda and Abejide Ade-Ibijola

Year: 2019

Description: The development of PathBot presented as a tool to assist students and visitors to navigate through the campus venues.

Limitations: It does not handle rerouting and cases where the user is lost.

3. Speech Recognition using Deep Neural Networks

Authors: Ali Bou Nassif, Ismail Shahin, mtinan Attili, Mohammad Azzeh, and Khaled Shaalan.

Year: 2019

Description: Automatic speech recognition is the capability of a machine to recognize the content of words and phrases in an uttered language and transform them to a machine-understandable format.

Limitations: The discrimination between two languages may become challenging as well as accent recognition.

4. Covid-19: Deep Learning Application

Author: Connor Shorten, Taghi M Khoshgoftaar and Borko Furht.

Year: 2019

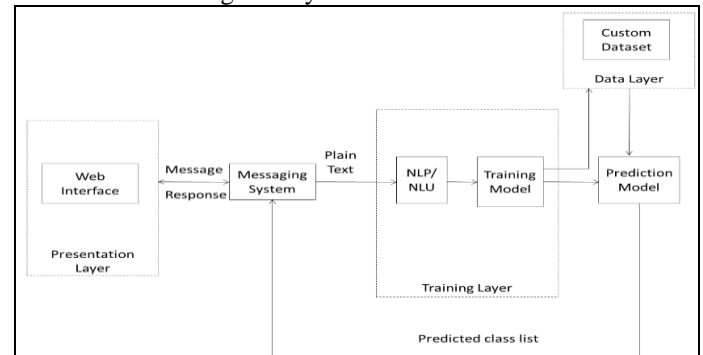
Description: This survey explores how Deep Learning has battled the COVID-19 pandemic and provides direction for future research on COVID-19.

Limitations: Interpretability, Generalization Metrics, Learning from Limited Labeled Data, and Data Privacy.

III. SYSTEM DESIGN

Initially, gather data from the dataset, which contains set of messages and their corresponding responses. Then, remove punctuations, extra symbols and other unnecessary things from this dataset. Then apply Bag of words technique to identify the class of input message. Feed the processed input query/message to Neural Network or Sequential model, which used to train chatbot. After this, predict the class and return the most appropriate response to user. This chatbot fetches the responses from dataset by matching the input with the conversations.

Diagram: System Architecture



IV. METHODOLOGY

1. Bag of Words

Bag of words model is the technique of pre-processing the text by converting it into a number/vector format, which finds the frequency of a word in a given sentence. It is the method of extracting features from the given text and use these features for model training.

• BAG OF WORDS MODEL:

This algorithm used for finding the frequency of a word in a given sentence. It is a way of extracting features from the given text.

a) To understand this, let us take an example:

“I have fever.”

“I have cough.”
 “I am feeling runnynose.”
 “I am having shortness of breath.”

b) We divide each sentence into words by neglecting punctuation marks.

Today, is, Sunday, a, holiday, it, chatbot, I, like, horse

c) Next, we should create tensors from the data. These vectors are used to process the data. Let us check the frequency of words in the first sentence

- “I” = 1
- “have” = 1
- “fever” = 1
- “cough” = 0
- “am” = 0
- “feeling” = 0
- “runnynose” = 0
- “having” = 0
- “shortness” = 0
- “of” = 0
- “breath” = 0

d) Later the documents will be represented as shown below:

“I have fever” = [1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0]
 “I have cough” = [1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0]
 “I am feeling runnynose” = [1, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0]
 “I am having shortness of breath” = [1, 0, 0, 0, 1, 0, 0, 1, 1, 1, 1]

➤ MATRIX REPRESENTATION OF BAG_OF_WORDS MODEL: -

Documents	I	H	F	C	A	F	R	H	S	O	B
	a	a	e	o	m	e	u	a	h	f	r
	v	v	v	u		l	n	a			
	e	e	e	g		l	o	h			
1	1	1	1	0	0	0	0	0	0	0	0
2	1	1	0	1	0	0	0	0	0	0	0
3	1	0	0	0	1	1	1	0	0	0	0
4	1	0	0	0	1	0	0	1	1	1	1

2. Sequential model

Create a sequential model. It is the simplest method to build a model in Keras. It allows to build a model layer by layer (starting from input layer to output layer).

Some of the important concepts related to sequential model are described as follows:

1. *Dense* is the type of layer used. In a dense layer, all nodes in the previous layer are connected to the nodes in the current layer.
2. *Activation* function allows model to represent non-linear relationships. The activation function is ReLU (Rectified Linear Activation) and SoftMax.

3. Predict class

Find the class with the highest probability and return the corresponding response.

V. IMPLEMENTATION

1) Create Intents

An “intent” is the intention of the user interacting with a chatbot or the intention behind each message that the chatbot receives from a particular user [9]. Define simple intents and set of messages that corresponds to those intents and also map some responses according to each intent category. Create a JSON file named “intents.json”, which is in the form of categories provided with some tags. Define set of patterns with the corresponding responses. Following is the sample dataset:

Figure 1: Create Intents

```

{
  "intents": [
    {
      "tag": "greetings",
      "patterns": ["Hi", "hello", "whats up", "hey", "Hi there", "hey there", "Greetings"],
      "responses": ["Hi/How may I help You?", "Hello/How may I help You?", "How may I help You?", "Hey, How you Doing today?"]
    },
    {
      "tag": "introduction",
      "patterns": ["What is your name", "May i know your name", "what can i call you", "What is your good name"],
      "responses": ["I am a CureBot.", "My name is CureBot, Tell me how can i help you?"]
    }
  ]
}
    
```

2) Data Preparation

Load the json file and extract the required data.

Figure 2: Data Preparation

```

intents = json.loads(open('intents.json').read())
words = []
classes = []
documents = []
ignore_letters = ['?', ',', '.', '!']

for intent in intents['intents']:
    for pattern in intent['patterns']:
        word_list = nltk.word_tokenize(pattern)
        words.extend(word_list)
        documents.append((word_list, intent['tag']))
        if intent['tag'] not in classes:
            classes.append(intent['tag'])
    
```

The variable “words” holds all the tokenized training data (which are sample messages in each intent category/tag), the “classes” variable holds all target labels/tags correspond to each training data and the “documents” variable holds both training data (words of sample messages tokenized using “word_tokenize(pattern)”) and labels correspond to each training data. After this, eliminate symbols, defined in “ignore_letters” variable, from training data and also eliminate stop words, if required.

3) Create Bag of Words

Convert text data into numeric format (0 and 1). Create bag of words, such as if word is present in “word_patterns” append 1, otherwise append 0 to the list.

Figure 3: Bag of Words

```
for document in documents:  
    bag = []  
    word_patterns = document[0]  
    word_patterns = [Lemmatizer.lemmatize(word.lower()) for word in word_patterns]  
    for word in words:  
        bag.append(1 if word in word_patterns else bag.append(0))
```

4) Model Training

Define Neural Network architecture for proposed model and for that use the “Sequential” model class of Keras. Keras is an open source, high level library for developing neural network model [10].

The steps for creating a Keras (Sequential) model are the following:

Step 1: Define neural network model (i.e., Sequential model). The network is defined as a sequence of layers, each with its own customizable size and activation function. In this sequential model, the first layer is the input layer, which define the size of the input, which is feed to neural network. After this more and more layers (hidden layers) can be added and customized until reached to the final output layer

Figure 4: Model training (step:1)

```
model = Sequential()  
model.add(Dense(128, input_shape=(len(train_x[0]),), activation='relu'))  
model.add(Dropout(0.5))  
model.add(Dense(64, activation='relu'))  
model.add(Dropout(0.5))  
model.add(Dense(len(train_y[0]), activation='softmax'))
```

Step 2: Define the optimization algorithm that will be used to train the sequential model, for that use “SGD” optimizer and also choose the loss function. Then, compile the neural network, which transforms the simple sequence of layers into a complex group of matrix operations that describes the behavior of the network.

Figure 5: Model training (step:2)

```
sgd = SGD(lr=0.01, decay=1e-6, momentum=0.9, nesterov=True)  
model.compile(loss='categorical_crossentropy', optimizer=sgd, metrics=['accuracy'])
```

Step 3: Train or fit the neural network and save it in “chatbot_model.h5” file.

Figure 6: Model training (step:3)

```
hist = model.fit(np.array(train_x), np.array(train_y), epochs=500, batch_size=5, verbose=1)  
model.save('chatbot_model.h5', hist)
```

Step 4: Now, model/network is trained, an input can be used to make prediction

5) Get User Input

Accept input query/message from user with developed frontend using Flask framework.

6) Predict Class for Input

Filter out the class/tag for user input, with specified threshold and make list of all filtered tags. Sort the list in reverse order, so that it returns class/tag, which has the highest probability. Finally, return the list, with classes and their corresponding probability.

Figure 7: Predict Class

```
def predict_class(sentence, model):  
    # filter out predictions below a threshold  
    p = bow(sentence, words, show_details=False)  
    res = model.predict(np.array([p]))[0]  
    ERROR_THRESHOLD = 0.25  
    results = [(i, r) for i, r in enumerate(res) if r > ERROR_THRESHOLD]  
    # sort by strength of probability  
    results.sort(key=lambda x: x[1], reverse=True)  
    return_list = []  
    for r in results:  
        return_list.append({"intent": classes[r[0]], "probability": str(r[1])})  
    return return_list
```

7) Get Bot Response

Store the class/tag, returned by previous function, having the highest probability. Compare that class/tag, with tags defined in “intents.json” file. Return the response corresponding to that class/tag.

Figure 8: Get Bot response

```
def getResponse(ints, intents_json):  
    tag = ints[0]["intent"]  
    list_of_intents = intents_json["intents"]  
    for i in list_of_intents:  
        if i["tag"] == tag:  
            result = random.choice(i["responses"])  
            break  
    return result
```

8) Display Bot Response

Display bot Response using flask framework.

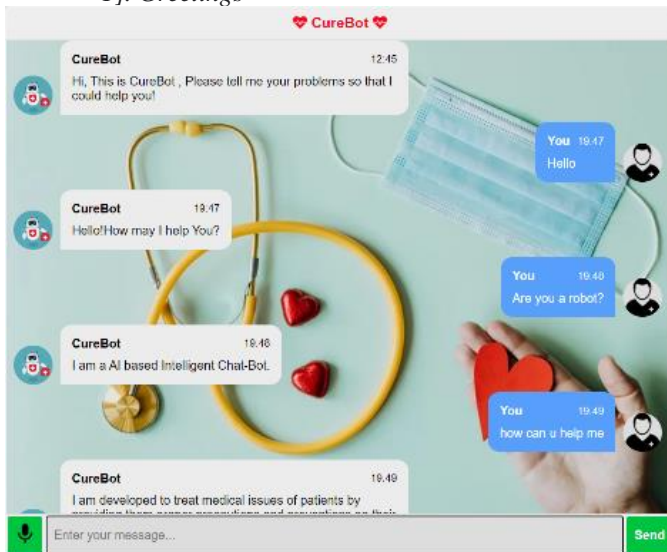
VI. RESULT

In this research we learned what **Artificial Neural Networks and Deep Learning** are. Some neural network makes the productive use of sequential data like text or audio which were introduced in Cure-Bot.

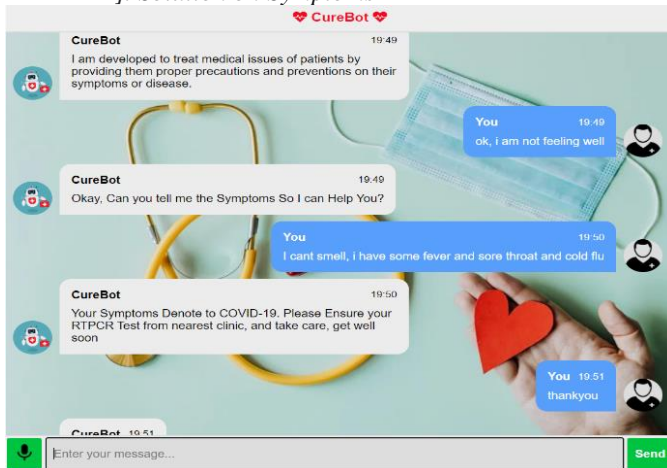
A deep learning chatbot learns right from scratch through a process called “Deep Learning” and using the data provided. In this process, the chatbot is created using machine learning algorithms. Some Framework and libraries used throughout the process are

- a. NLTK,
 - b. Keras,
 - c. NumPy,
 - d. TensorFlow.
 - e. Flask: made possible to build a web Application by providing tools, libraries and technologies.
- This are the following Screenshots provided from CureBot working model—>

1]. Greetings



2]. Solution on Symptoms



3]. End of chat



VII. CONCLUSION

In conclusion, we have presented an Artificially Intelligent Chat-bot using applications of Deep Learning to fight COVID-19 including various viral diseases faced by human being in day Today life. Keeping in mind, the situation of the rural population and imbalance between the demand and healthcare services currently provided, so by developing this CureBot will bridge a gap by creating a conversational application with natural language processing.

Our survey gives solution on how human can tackle various diseases implementing Deep learning in CureBot. We have shown how the Data is given as input to the Deep Neural Network and how task is constructed as learning problem. We have covered some solutions to the user’s query which will be beneficial for proper understanding of the patients. We also have implemented speech to text conversation type for better use of Chat-bot.

REFERENCES

- [1] Vaibhav Tode, Himanshu Gadge, Prateek Kachare and Sudarshan Madane, " CureBot -An Artificially Intelligent Interactive Bot for Medical Diagnostics" *International Research journal of Engineering and Technology (IRJET).*, Vol.7, no.12 (Dec 2020).
- [2] Satyendra Praneel Reddy Karri and Dr Santosh kumar, " Deep Learning Techniques for Implementation of Chatbots", 2020 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, INDIA, (2020) January 22-24.
- [3] Dipesh Kadariva, Revathy Venkataramanan, Hong Yung Yip, Maninder Kalra, Krishnaprasad Thirunarayanan and Amit Sheth, "kBot: Knowledge-Enabled personalized Chatbot for Asthma Self-Management", 2019 IEEE International Conference on Smart Computing (SMARTCOMP), Washington, DC, USA, (2019) June 12-15
- [4] Katlego Mabunda and Abejide Ade-Ibijola, " PathBot: An Intelligent Chatbot for Guiding Visitors and Locating Venues", 6th International Conference on Soft Computing & Machine Intelligence (ISCMi), Johannesburg Parktonian All-Suite, South Africa, (2019) November 19-20.
- [5] Ali Bou Nassif, Ismail Shahin, Imtinan Attili and Mohammad Azzeh, " Speech Recognition Using Deep Neural Networks: A Systematic Review" *IEEE-Access*, Vol. 7, (2019), pp(99).1-1.
- [6] Connor Shorten, Taghi M. Khoshgoftaar and Borko Furht, " Deep Learning applications for COVID-19", *SpringerOpen J Big Data* 8, Article no8, (Jan 2021).

- [7] *Eleni Adamopoulou and Lefteris Moussiades, "Chatbots: History, Technology, and applications", ELSEVIER Machine Learning with Applications, Vol. 2, (Dec 2020).*
- [8] <https://www.investopedia.com/terms/d/deep-learning.asp#:~:text=Deep%20learning%20is%20a%20subset,learning%20or%20deep%20neural%20network>.
- [9] <https://towardsdatascience.com/how-to-build-your-own-chatbot-using-deep-learning-bb41f970e281>
- [10] <https://towardsdatascience.com/deep-learning-for-nlp-creating-a-chatbot-with-keras-da5ca051e051>
- [11] *Deep Learning Definition – Investopedia.*
- [12] <https://towardsdatascience.com/how-to-create-a-chatbot-with-python-deep-learning-in-less-than-an-hour-56a063bdfc44>