CHATBOT FOR GOVERNMENT SCHEMES

PROJECT REPORT submitted in partial fulfillment of the requirements

Submitted by

LAKSHMI KEERTHANA ARAVAPALLI - 178W1A1201
GOWRINAGH GUDIVADA - 178W1A1218
DHRUTHI NARAGANI - 178W1A1237

Under the Guidance of

Dr. M. SUNEETHA, Ph.D.

Professor & Head

For the award of the degree

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

DEPARTMENT OF INFORMATION TECHNOLOGY V R SIDDHARTHA ENGINEERING COLLEGE
(AUTONOMOUS - AFFILIATED TO JNTU-K, KAKINADA)

Approved by AICTE &Accreted by NBA
KANURU, VIJAYAWADA-

7 ACADEMIC YEAR
(2020-21)

V.R.SIDDHARTHA ENGINEERING COLLEGE

(Affiliated to JNTUK: Kakinada, Approved by AICTE, Autonomous)

(An ISO certified and NBA accredited institution)

Kanuru, Vijayawada – 520007

CERTIFICATE

This is to certify that this project report titled **"CHATBOT FOR GOVERNMENT SCHEMES"** is a bonafide record of work done by **A.L.KEERTHANA (178W1A1201), G.GOWRINAGH (178W1A1218) and N.DHRUTHI (178W1A1237)** under my guidance and supervision is submitted in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Information Technology, **V.R. Siddhartha Engineering College** (Autonomous under JNTUK) during the year **2020-2021**.

Dr. M. Suneetha

Professor & Head

Dept. of Information Technology

Dr. M. Suneetha

Professor & Head

Dept. of Information Technology

TABLE OF CONTENTS

ACKNOWLEDGEMENT

First and foremost, I sincerely salute our esteemed institution ***V.R SIDDHARTHA ENGINEERING COLLEGE*** *for giving me this opportunity for fulfilling my project. I am grateful to our principal* ***Dr. A.V.RATNA PRASAD****, for his encouragement and support all through the way of my project.*

On the submission of this Project report, I would like to extend my honour to ***Dr. M.Suneetha****, Head of the Department, IT for her constant motivation and support during the course of my work.*

I feel glad to express my deep sense of gratitude to my project guide ***Dr. M.Suneetha Professor & Head of the Department, IT*** *for her guidance and assistance in completing this project successfully.*

I would also like to convey my sincere indebtedness to all faculty members, including supporting staff of the Department, friends and family members who bestowed their great effort and guidance at appropriate times without which it would have been very difficult on my part to finish the project work.

List of Figures

List of Tables

Abstract

Government is providing numerous schemes in different domains like education, agriculture, social-welfare, healthcare. Government schemes are always beneficial for the public, especially for low-income groups. But, sometimes it becomes difficult to understand these schemes and their benefits. Mostly people living in rural areas are not able to avail the benefits provided by the government through these schemes due to lack of awareness. Even though the information is existing in multiple websites on the internet, the challenge is, they need to go through multiple websites to get the required information. There are some existing apps and websites, but the information is limited to some areas. So the idea of this project is to build a chatbot that can answer queries related to some specific government schemes. In this application anyone who wants to know about the schemes provided by the central government in the domain of social welfare and girl child can ask their query to the chatbot and can easily get the answers from the chatbot. The chatbot also includes a speech feature so the user can interact with speech. The chatbot application also supports 2 languages Telugu and English. To build the chatbot model we use Convolutional neural Networks and Natural language Processing. A fundamental piece of machinery inside a chat-bot is the Text Classifier and Convolutional Neural Networks have found prevalence in issues related to human language technology tasks like Text Classification. This chatbot can help people to know about the beneficial schemes provided by the central government and get the information related to the application process, required documents, eligibility, where to apply and last date to apply.

Keywords - Government Schemes, Chatbot, DeepLearning, Convolutional Neural Network, Natural Language Processing.

Chapter 1

Introduction

Many schemes are being offered to the people by the government in the domain of education, agriculture, social welfare, healthcare. Government schemes are always beneficial for the public, especially for low-income groups. But people mostly those living in rural areas out of unknownness or due to lack of awareness over the schemes are not able to claim these schemes. Sometimes people don't know the schemes that are available for them or to which schemes they are eligible. so this chatbot can help them know about these schemes and can avail the benefits provided by them. So the main idea of this project is to build a chatbot using Deep Learning that provides information on a few schemes related to social welfare and girl child provided by the central government.

A chatbot is a conversational interface software program that allows a user to converse in the same way as one would address a human. A virtual chatbot is a software smart enough to mimic human interactions. Chatbots are used in almost all customer interactions, like instantly sending messages to the customer. Deep learning and NLP techniques are used in advanced research and development projects, and AI and ML algorithms are used in conversation development. Conversation agents are mainly used by public administrations, businesses, and non-profit establishments. For our chatbot, we collected the data related to the schemes and trained the model using the data. In our chatbot, we include schemes related to Social Welfare and Girl Child. This chatbot provides details like schemes available, eligibility, necessary dates, and process to apply to a particular scheme.

1.1 Origin of the Problem

On interacting with some people randomly and had a general conversation with them regarding the government schemes, we came to know that some of them don't even know about the schemes and some of them know the scheme but they don't know how to avail the benefits of it. If they want to know about them there are only limited resources in certain languages. So we want to make our chatbot in such a way that they can converse in either telugu or english so that there is no language problem and if they can't type they can use the speech option which helps them to converse.Our chatbot helps society in a well-versed way where the users will feel satisfied with the information

they obtained from our chatbot. The main aim of our chatbot is to help users with the doubts regarding the government schemes.

1.2 Basic Definitions

A. *Chatbot:* :

A **chatbot is** a computer program that simulates human conversation through voice commands or text chats or both. **Chatbot**, short for chatterbot, **is** an Artificial Intelligence (AI) feature that can be embedded and used through any major messaging applications. Chatbots are used in dialog systems for various purposes including customer service, request routing, or information gathering.

B. Artificial Intelligence:

Artificial Intelligence (AI) is the branch of Computer sciences that emphasizes the development of intelligent machines, thinking and dealing like humans. For example, speech recognition, problem-solving, learning, and designing.

C. Natural Language Processing:

Natural language processing (NLP) is a collective term referring to the automatic computational processing of human languages. This includes both algorithms that take human-produced text as input and algorithms that produce natural-looking text as outputs. The NLP developers can organize the knowledge to perform various tasks such as analyze text, text summarizing, topic extraction, stemming, text mining, automatic summarization, translation, speech recognizing, segmentation, and automated question answering.

D. *Keras*:

Keras is a neural networks library written in Python that is high-level in nature – that makes it very simple and intuitive to use. It works as a wrapper to low-level libraries like TensorFlow or Theano high-level neural networks library, written in Python that works as a wrapper to TensorFlow .

E. *Tkinter*:

Tkinter is a Python binding to the Tk user interface toolkit. it's the quality Python interface to the Tk user interface toolkit and is Python's standard user interface. Tkinter is enclosed with the

standard UNIX system, Microsoft Windows and mackintosh OS X installs of Python. The name Tkinter comes from the Tk interface.

F. *TensorFlow*:

It is an open-source artificial intelligence library, using data flow graphs to build models. It allows developers to create large-scale neural networks with many layers. TensorFlow is mainly used for: Classification, Perception, Understanding, Discovering, Prediction and Creation.

G. *Application Programming Interface(API)*:

API stands for Application Programming Interface. It acts as an intermediate between two applications or software. In simple terms, API acts as a messenger that takes your request to destinations and then brings back its response for you. For example, Google provides APIs for speech, language translation, and language detection.

1.3 Problem statement

Our government is providing so many schemes in different areas like agriculture, health, education, girl child, social welfare, energy.These schemes are for the welfare of the people.But most of the people mainly those living in rural areas are not getting benefitted from these due to lack of knowledge on them. So we create a chatbot using AI that helps them to get information like schemes available, what that scheme mean, who are eligible for it, how to apply and the process

1.4 Applications

Chatbot can be used to provide information related to government schemes to illiterate people.
This chatbot can be used by the students who are preparing for exams.
The artificial intelligence-based conversational platform is expected to interact with the public in multiple languages, analyse sentiments and intent, collect and analyse data to give personalised experience to users.

Chapter 2

Review of Literature

2.1 Description of Existing Systems

There are some existing apps and websites that provide information related to government schemes. Following are some of the Existing apps and websites

Existing Apps:

- My Government Schemes: This app provides information on all schemes according to the ministry. The information includes the objective, description, and eligibility of the scheme.
- All Government Schemes: This app gives information on 31 Schemes introduced by PM Narendra Modi. The information includes Description, Eligibility, Documents Required to apply, Procedure of Applying.
- Government Schemes: It contains All Schemes of Central Government and Some State Governments schemes. All information related to the scheme is provided. The information is provided in the Hindi language.
- Gov Info: It gives a description of all central and state government schemes. The information includes only a description.

Websites:

- https://rural.nic.in/scheme-websites: This website provides some of the rural development schemes and their respective links. There is no information regarding the process to apply and only confined to 15 schemes.
- https://byjus.com/govt-exams/government-schemes/: This website provides some of the schemes provided by the Government of India related to Poverty Alleviation, Employment, Health, and sanitization.
- https://www.startupindia.gov.in/content/sih/en/government-schemes.html: It provides 125 schemes in which we can select the schemes Ministry and Department but only related to startups under each ministry.
- http://socialjustice.nic.in/SchemeList/index?mid=24541: It provides the schemes regarding Economic Development and Social Empowerment. These schemes cover all scholarships

and some schemes on education.

Journals:

- Programming challenges of Chatbot: Current and Future Perspectives This document was submitted by AM Rahman, Abdullah Al Mamun, Alma Islam in 2017 IEEE Region 10 Humanitarian Technology Conference [3]. He provided an overview of chatbot Technologies and programming challenges in the current and future era of the chatbot. We deduce that the dynamic response using the knowledge base provides better results than the static response. Programming challenges include NLP and Machine Learning.

- A scaled-down neural conversational model for chatbots: This paper was submitted by Saurabh Mathur, Daphne Lopez in 2018 John Wiley & Sons, Ltd. It provided an overview of different neural network models that can be used to develop chatbots. It also describes the encoder, decoder architecture, and experimental setup for RNN. It is demonstrated that bidirectional decoders perform better than unidirectional ones, on the task of short text conversation modeling. The use of the ELU activation function instead of the function is proposed, and the ELU activation function is shown to improve the performance of Gated Recurrent Units on conversation modeling.

2.2 Summary of Literature Study

Most of the apps and websites are providing information like description of the scheme, eligibility and required documents. But the process of application is not being provided. Also some apps were developed in only one language which fails to serve the purpose if the user doesn't know the language. And also if we present the information in a website, users need to go through the whole website even for small questions.

So this project aims to develop a chatbot in multiple languages (Telugu, Hindi, English) using machine learning algorithm so the users can ask the question and get the answer for it without having to go through the entire website.

Chapter 3
Proposed Method

This chapter discusses about the proposed methodology and the algorithm used to develop the chatbot. It also describes the steps involved in implementing the chatbot.

3.1 Design Methodology

In this Project, the user asks the query in the form of text or speech to the chatbot, for the details of a particular scheme. The query is sent to the model which is already trained using predefined intents. Intents include information like scheme description, eligibility, required documents, and procedure to apply which acts as a database for a chatbot. The model checks if the query matches the chatbot database entry. If the exact entry is found in the database, a related response from the data is given by the chatbot. If it does not find the response in the database, the bot asks the user to train it accordingly. The response is displayed to the user in the form of both text and speech. The chatbot answers the user in the language in which the question is received (languages included are Telugu, English, Hindi). Figure 3.1 shows the UML representation of the project.

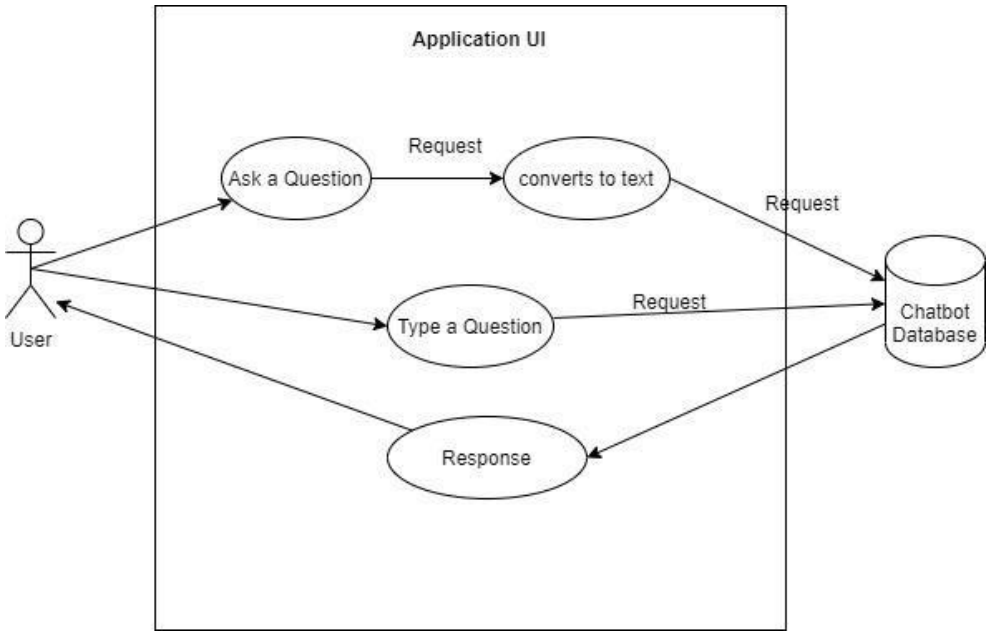


Figure: 3.1 UML Diagram

3.2 System Architecture Diagram

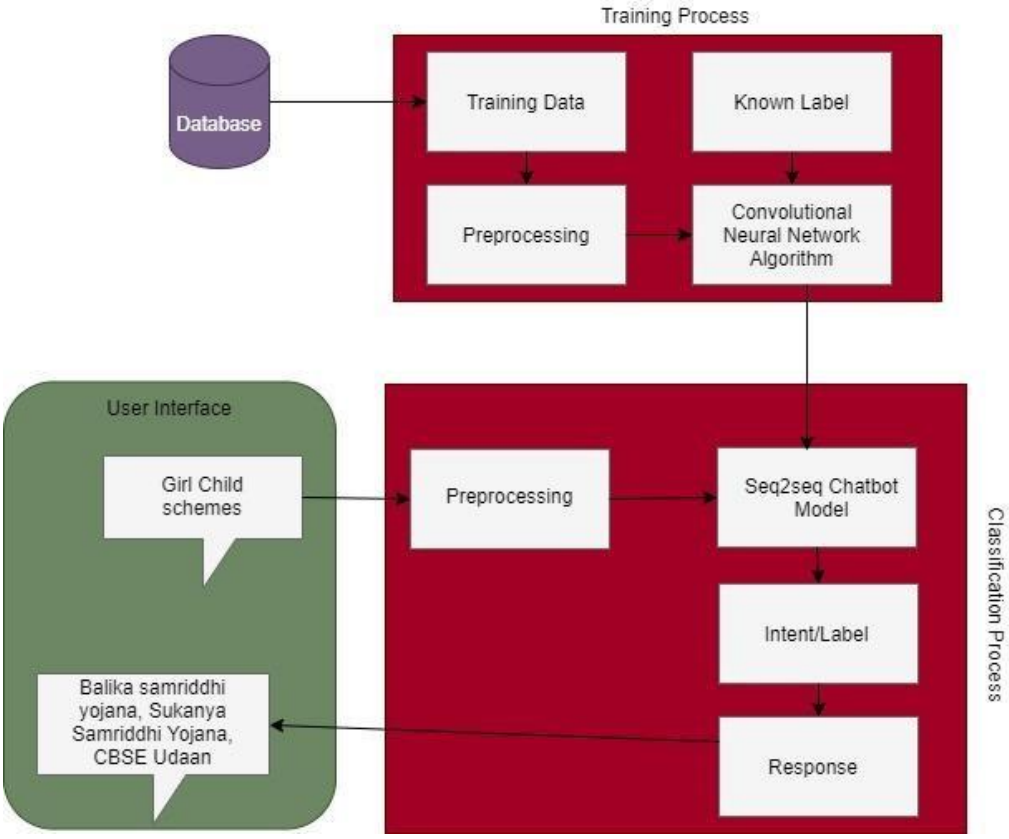


Figure: 3.2 Architecture Diagram

The architecture diagram of the project is shown in Figure 3.2. It can be described as follows Training Process:

The Training data consists of tags, patterns and responses. Preprocessing techniques like removing stop words, tokenization, lemmatization and vectorization are applied on the training data. In this project, the Convolutional Neural Networks model is used to develop the chatbot. Now the model will be trained using the predefined labels(tags) and preprocessed training data.

Tokenization:

Tokenization is a way of separating a piece of text into smaller units called tokens. Here, tokens can be either words, characters, or subwords. Hence, tokenization can be broadly classified into 3 types – word, character, and subword (n-gram characters) tokenization.

Lemmatization:

In this process, different inflected forms of a word are put together in the form of a group. The output obtained is a perfect word after combining all the words.

Vectorization:

Word vectorization is a methodology in NLP to map words or phrases from vocabulary to a corresponding vector of real numbers which used to find word predictions, word similarities/semantics.

CNN:

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in input, assign importance (learnable weights and biases) to various aspects/objects in the input and be able to differentiate one from the other.

Classification Process:

When the user enters the query, the query will be preprocessed and will be sent to model to generate an appropriate response. The model classifies the query into one of the tags present in the training data and gives the response from the respective tag.

3.3 Overview of Algorithm

The algorithm used in building this chatbot is CNN(Convolutional Neural Networks) which is used to classify the category for which the user's message belongs to and it will give an arbitrary response from the record of responses using text classification. A fundamental piece of machinery inside a chat-bot is the *text classifier*. Convolutional Neural Networks have found prevalence in issues related to human language technology tasks like Text Classification.

A convolutional neural network consists of “convolutional” layers and “downsampling” or “subsampling” layers. Convolutional layers comprise neurons that scan their input for patterns. Downsampling layers or “pooling” layers are typically placed when convolutional layers in an exceedingly ConvNet, chiefly to cut back the feature map spatial property for machine potency,

which might successively improve actual performance. This is often followed by associate degree MLP with one or a lot of layers(totally connected layer). We represent the text as an associate degree array of vectors(every word mapped to a selected vector in an exceeding vector area composed of the complete vocabulary) which will be processed with the assistance of a CNN. The following Figure 3.3 represents the General Architecture of a CNN model for Text Classification.

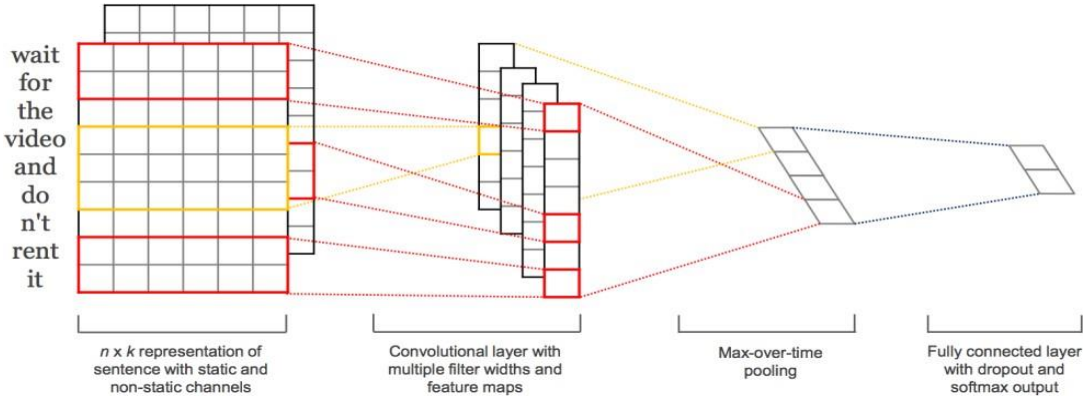


Figure: 3.3 Convolutional Neural Network Architecture[3]

The implemented Convolution Neural Network model has six layers- Embedding layer, pooling layer, 3 hidden layers and one fully connected layer. In text classification tasks CNN gives promising results. Text classification is similar to image classification. The only difference is that instead of pixel values we have a matrix of word vectors. The proposed model is implemented in python using the TensorFlow library.

Target performance used includes neurons with learnable weights and biases. Every nerve cell receives many inputs, and weights added over them, activation function is applied on them and responds with an output. The whole network contains a loss performance calculated by spreading output to the softmax layer. Softmax could be a totally connected layer that conjointly performs downsampling for the output.

The first layer in convolutional neural networks is the embedding layer that maps vocabulary word indices to low dimensional vectors. It's primarily a search table that has a tendency to learn from knowledge. Vocabulary size is determined by obtaining the most sentence length. Given a sentence of N words W each word is remodeled into its various embeddings.

After the transformation of all the words to vectors, these are then fed to the convolution layer. The embedding lookup function is used to get the word embedding of the sentence. The matrix generated as a result of the embedding layer is then cushioned to equalize all the sentences. The outlined filters can then begin reducing the matrix and generate convolved options. These convolved options are then reduced. The output generated as a result of convolved options is then touched by the gamma-hydroxybutyrate pooling layer for additional downsampling of output.

The pooling operation is employed to mix the vectors ensuing from completely different convolution windows into one-dimensional vectors. This is often done once by taking the typical worth ascertained in the ensuing vector from the convolutions. Ideally, this vector can capture the foremost relevant options of the sentence/document.

Filters of various sizes and shapes square measure outlined. The filters can rollover the first sentence matrix, therefore, reducing it to a low dimensional matrix. . Instead of training our own embedding, we use the embedding lookup function of TensorFlow. Embedded sentences are then padded to make all the sentence matrices of the same size and shape. In Data processing by using Natural Language Processing techniques features are selected from the text. These features will be then used to categorize the text. Feature selection is an important phase because the categories defined in this phase will be used for classification purposes. The following Figure 3.4 represents the General Architecture of a CNN model for Text Classification.

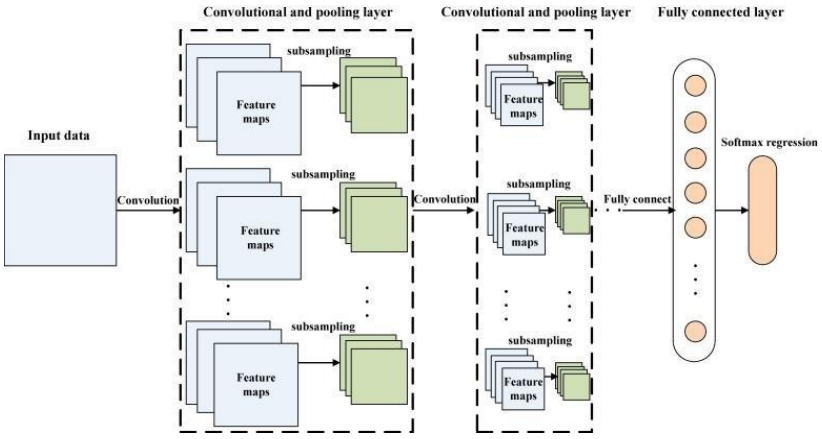


Figure: 3.4 Pooling layers in CNN[6]

The proposed chatbot also employs word mappings. Mappings are used to define the word

embedding. Obtained categories of text along with related text and their respective word embedding are used to classify text using the proposed Convolution Neural Network Model. Neural Network Model will develop its knowledge base.

Once the classification model is trained on the information, the same model can be used to get a response to the questions. Question is tokenized and tagged using natural language processing techniques. Questions will be classified into categories. Once the successful classification of the question occurs, the system will compare the semantic information of related category text to the question. By comparing semantic information of a question to related category text in which question is classified answer selection occurs. The system will also add new semantic information to the knowledge base so that it can learn and handle diverse questions.

Algorithm steps

Table 3.1 describes the steps involved in CNN algorithm.

Table 3.1 Algorithm Steps

StepNo.	**Description**
1	Input: Preprocessed training data containing tags, patterns and responses.
2	The first layer is the embedding layer with vocab_size 1000, embedding dimension 16, input_length 20.
3	Second layer is the global average pooling 1D layer
4	Use 3 dense layers with 64,16,16 units respectively with activation function 'relu'.
5	Last layer of model will be another Dense Layer, with units equal to number of tags.
6	A.F = Softmax.
7	The softmax function outputs a vector which will represent the probability distributions of each of the input units.
8	Predict the tag.
9	End

3.4 Description of Dataset

The data is gathered from trustable resources like government websites. The dataset contains many tags and every tag has patterns and responses. These patterns and responses take questions and answers respectively. The dataset incorporates a wordbook mapping of dictionaries. First, intents and mapped with tags, patterns, context, and responses, then every one of them is mapped with their queries and keywords. Table 3.2 describes the attributes in the dataset.

Table: 3.2 Dataset Description

S. No	**Attribute Name**	**Description**
1.	Tags	Tags are keywords appointed to the patterns and responses for coaching the chatbot. e.g., “greeting”, “eligibility”
2.	Patterns	These are the categories of queries asked by the user to the chatbot. e.g., “Hello”, “Hi there”
3.	Responses	These are the responses from which the model has to choose to answer.

3.5 API’s Used

Google Speech Recognition API:

Google has a great speech Recognition API. This API converts spoken text(microphone) into written text (python Strings), briefly Speech to Text. Users can simply speak on a microphone and Google API will translate this to written text.

Language Translation:

The chatbot needs to take input in both Telugu and English. So when the user gives the input in Telugu bot needs to convert it into English so that it matches the training data. In order to present the results in the required language,google translator API named “**googletrans**” is used. This API is used to convert text from one language to another language.

Language Detection:

The chatbot needs to understand the language in which the query is being asked and should return the response in the same language. To achieve this, "**langdetect API"** from google language detection library is used.

3.5 Implementation

This project uses Deep Learning techniques like CNN to build the chatbot. The chatbot is trained with the government schemes dataset that comprises intents, patterns, and responses. The libraries that are imported to create government schemes chatbot are NLTK, TensorFlow, Keras, sklearn.

Steps to build the chatbot:

1. Import required libraries and load the data:

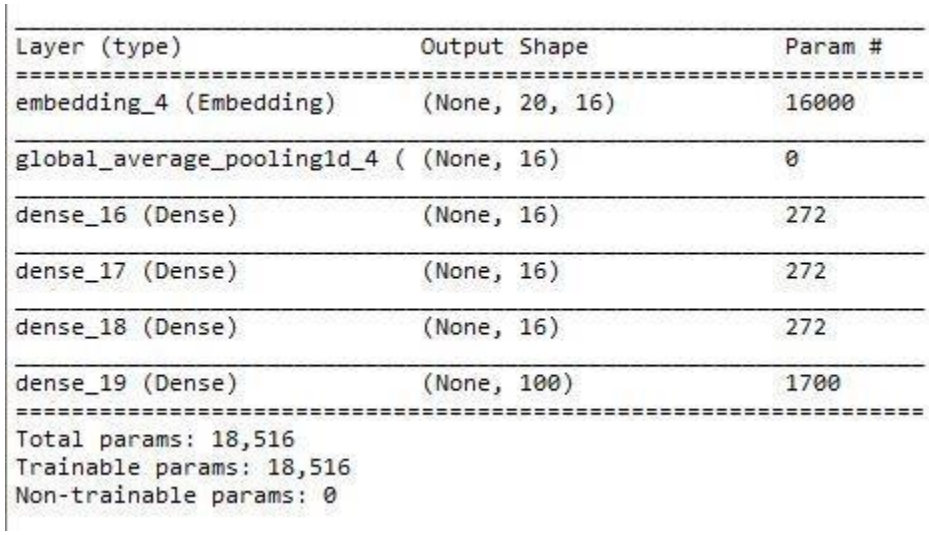
Create a python file named train_chatbot where all the required modules and packages are imported from the JSON package to read the JSON data file into the python program.

2. Preprocessing the data:

Data preprocessing is the step during which information gets reworked or encoded to bring it to such a state that the machine can easily parse it. In other words, the features of the data can now be easily interpreted by the algorithm. The label encoder provided by scikit-learn is used to convert the target labels into a model understandable form. To vectorize our text information corpus, the "Tokenizer" class is used. It permits to limit the vocabulary to some value. After this class is used for the text pre-processing task, by default all punctuations are going to be removed, turning the texts into space-separated sequences of words, and these sequences are then split into lists of tokens. They're going to then be indexed or vectorized. Additionally a "oov_token" is added that may be a price for "out of token" to traumatize out of vocabulary words(tokens) at inference time. The "pad_sequences" methodology is employed to form all the training text sequences into constant size.

Training the model:

Deep learning model's architecture that is used in this model consists of an embedding layer, global average pooling layer,3 dense layers and one fully connected layer. The Keras sequential API is used for this. The first layer converts the text into a numerical array of size 16. The average pooling layer extracts the features from the training data. The first dense layer consists of 128 neurons; the second and third layer has 64 neurons. ADAM optimizer is used as an optimizer and the model is trained using the data. After the training of 500 epochs is finished, the model's accuracy reaches 96%, then the model is saved using Keras model.save ("chatbot_model.h5") function. Figure 3.5 shows the model summary of the CNN model.



Layer (type)	Output Shape	Param #
embedding_4 (Embedding)	(None, 20, 16)	16000
global_average_pooling1d_4 (	(None, 16)	0
dense_16 (Dense)	(None, 16)	272
dense_17 (Dense)	(None, 16)	272
dense_18 (Dense)	(None, 16)	272
dense_19 (Dense)	(None, 100)	1700

Total params: 18,516
Trainable params: 18,516
Non-trainable params: 0

Figure: 3.5 Model Summary

Creating GUI:

Now create a new file named chatbot_gui to develop a user interface for the users to interact with the chatbot. Use tkinter module to develop the structure for the desktop application. Then the user message is received on which the preprocessing techniques are performed before the message is given as input to the trained model. The model then takes the input and will predict the tag of the user's message and will randomly select the response from the list of responses available in the intents file. The user interface is developed in such a way that the user can interact with the chatbot using text or speech in either English or Telugu.When the user asks the query through text it will be directly sent to the model. After getting the response from the model response is given to the user in the form of text and audio. When the user asks the query through speech it is converted into text and will be sent to the model. After getting the response from the model the response

will be given to the user in the form of audio. For text based input, the language of the response generated depends on the language of the user input. For speech the user selects the language by saying Telugu for telugu language and English for english language. After selecting the language user can converse with the chatbot in the selected language.

Chapter 4

Results and Observations

4.1 Files required for chatbot

A) Intents.json – This file consists of all the data that we make use of for training the model. The data file contains a collection of tags that have already defined patterns, responses in it.

B) train_chatbot.py – This file comprises a script that is used in building the model. It trains the model by using deep learning techniques to classify and identify what the user is asking the bot.

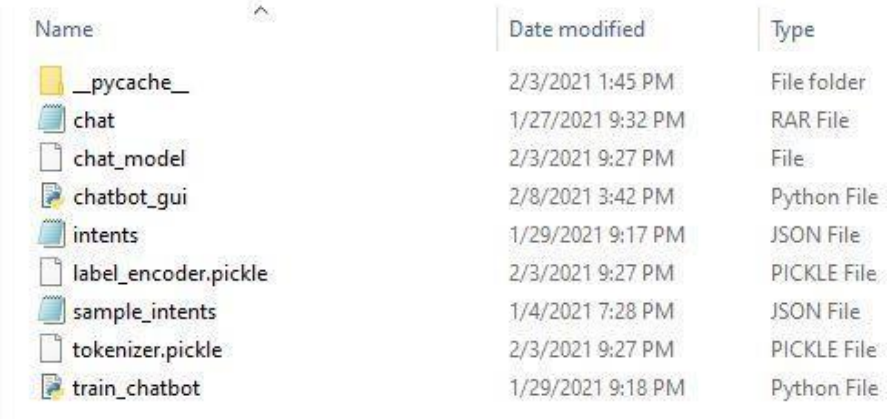
C) words.pkl – All the distinct words are stored in this pickle file and it comprises the records of our vocabulary.

D) Classes.pkl – The classes.pkl is also a pickle file that comprises the list of all categories that can be used to store all the tag names to classify when we are predicting the message.

E) Chatbot_model.h5 – This model is trained which comprises a hierarchical data format and has stored weights and the architecture of our training model.

F) Chatbot_gui.py – The Chatgui.py is a script file which will build a GUI (Graphical user interface) that is required for the better user experience to use the chatbot. This file consists of a script that can call the chatbot model and return the response for the user query.

Files required for this project are shown in figure 4.1.



Name	Date modified	Type
__pycache__	2/3/2021 1:45 PM	File folder
chat	1/27/2021 9:32 PM	RAR File
chat_model	2/3/2021 9:27 PM	File
chatbot_gui	2/8/2021 3:42 PM	Python File
intents	1/29/2021 9:17 PM	JSON File
label_encoder.pickle	2/3/2021 9:27 PM	PICKLE File
sample_intents	1/4/2021 7:28 PM	JSON File
tokenizer.pickle	2/3/2021 9:27 PM	PICKLE File
train_chatbot	1/29/2021 9:18 PM	Python File

Figure: 4.1 Files Required

4.2 Step-wise Description of Results

- An application is displayed to the user in which the user can type his query in the textbox and click on send.
- The response will be given by the bot to the query and is displayed to the user.
- The user can click on the speech button to interact with the bot through speech.
- The language in which the user wants to interact can be changed by saying the required language.
- In case of text the language is automatically detected by the bot.

4.3 TestCaseResults

The chatbot is tested with the following conversations. Testcase-1: First query is given to the chatbot in the form of text through the text box. When the query is asked about PMMVY in English, it gives the response in english. Then when asked about sukanya samriddhi yojana in telugu language it gives the response in telugu. The results of this testcase are shown in Figure-4.2.

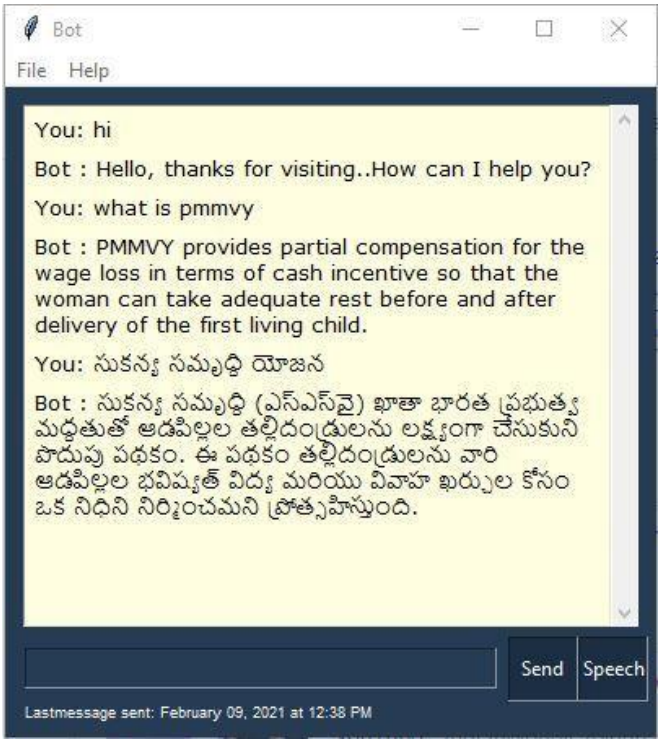


Figure: 4.2 Test-case Result

Testcase-2:Similarly when clicked on speech button, speech will be enabled so the user can ask the question through speech instead of typing it. When asked about Balika samriddhi yojana eligibility it gives the response in english as it is the default language. The results of this testcase are shown in Figure-4.3.

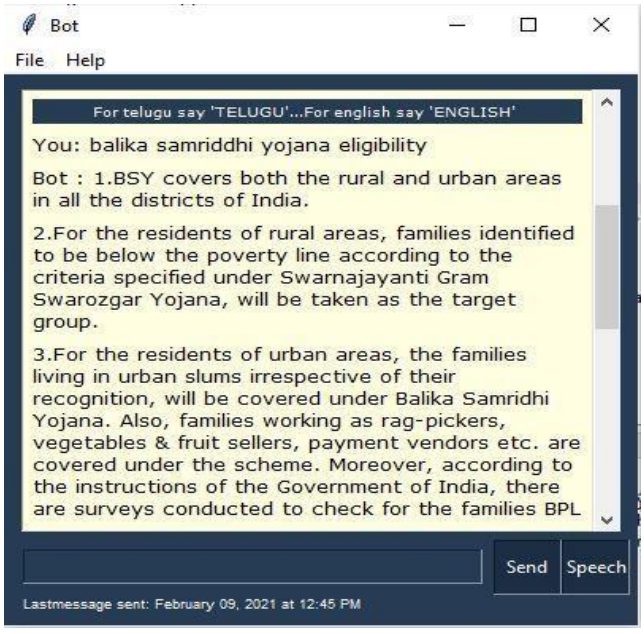


Figure: 4.3 Test-case Result

Testcase-3:If the user wants to change the language to telugu, they just need to say telugu and the language gets changed. After changing the language when asked about atal pension yojana it gives the response in telugu language. Users can say quit if they want to stop speech recognition. While getting the output users will get it in the form of both text and audio irrespective of the type of input. .The results of this testcase are shown in Figure-4.4.

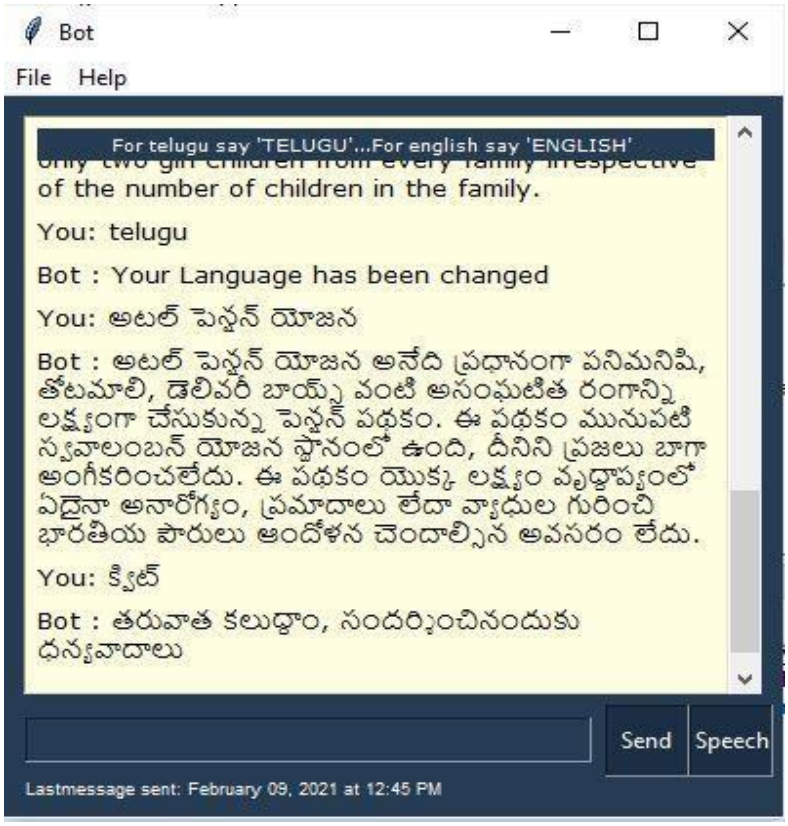


Figure: 4.4 Test-case Result

When the chatbot is opened the user can enter their query in the form of text or speech and can get the results in both text form and as audio. User can interact with the chatbot in either telugu or English based on his convenience. The chatbot requires good internet connectivity to interact with the user.

Chapter 5

Conclusion and Future Study

5.1 Conclusion

The main objective of the paradigm is to prosper a chatbot. This idea was developed by considering the problems faced by Common people. This chatbot is developed using CNN algorithm. It classifies the query based on the user input. API's are used to include speech as communication medium and to translate the text into desired language. This can help user to interact with the bot in either English or Telugu according to his choice. Tkinter toolkit is used to develop the GUI for the chatbot which enables the users to interact with the chatbot. This application can be used by anyone who wants to know about the schemes provided by the central government in the domain of social welfare and girl child can ask their query to the chatbot and can easily get the answers from the chatbot. So it can be concluded that a chatbot for government schemes will help the people in answering their queries to the point and quickly in their preferred language and medium.

5.2 Future work

This application can be extended by adding authorization services. Databases can be initiated to perpetuate users. More number of schemes in different sectors can be added. Application can be enlarged to web support. More number of languages can be added.

References

[1] Saurabh Mathur, Daphne Lopez "A scaled-down neural conversational model for chatbots"Concurrency Computat Pract Exper. 2018 Wiley.

[2] Camilo Thorne 'Chatbots for troubleshooting: A survey' Lang Linguist Compass. 2017 Wiley.

[3] Masters of Science, Computer Science, University of Memphis, Dr. Vasile Rus, Computer Science, University of Memphis, Intelligent Chatbot - Generative Open Domain Chatbot Application with Deep Learning [Python, TensorFlow, Seq2Seq, PyQT] Science Direct.

[4] Programming challenges of Chatbot: Current and Future Perspectives This document was submitted by AM Rahman, Abdullah Al Mamun, Alma Islam in 2017 IEEE Region 10 Humanitarian Technology Conference.

[5] Winograd, T. (1972). Understanding natural language. Cambridge, MA: Academic Press.

[6] van Schooten, B., and op den Akker, R. (2011). Vidiam: Corpus-based development of a dialogue manager for multimodal question answering.

[7]RNN or Recurrent Neural Network for Noobs – RNN Architecture https://hackernoon.com/rnn-or-recurrent-neural-network-for-noobs-a9afbb00e860

[8] Lisa N.Michaud, "Observations of a new chatbot – Drwaing conclusions from early interactions with users", Feature Article: Virtual Assistant Chatbots, published by IEEE Computer Society 2018.

[9] Rupesh Singh, Harshkumar Patel, Manmath Paste, Nitin Mishra, Nirmala Shinde, "Chatbot using TensorFlow for small Businesses", 2nd International Conference on Inventive Communication and Computational Technologies (ICICCT 2018)

[10] Shafquat Hussain , Prof. Athula Ginige, "Extending a conventional chatbot knowledge baseto external knowledge source and introducing user based sessions for diabetes education", 2018 32nd International Conference on Advanced Information Networking and Applications Workshops.

[11]Pure Python vs NumPy vs TensorFlow Performance Comparison
mini project

[12] Training a model https://pythonprogramming.net/chatbot-deep-learning-python-tensorflow/

[13] Nielson Norman Group "The user experience of chatbot" https://www.nngroup.com/articles/chatbots/.