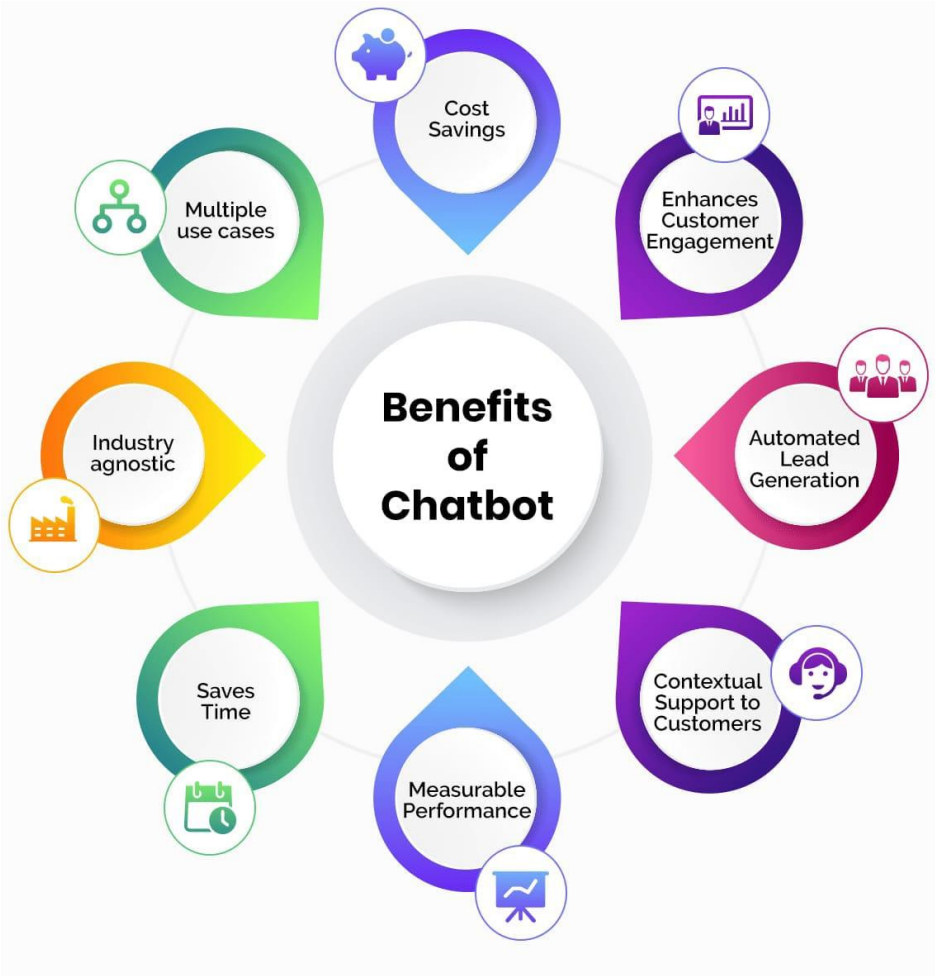
ARTIFICIAL INTELLIGENCE

CREATE A CHATBOT IN PYTHON

Phase 2 : Innovation

- ✓ ChatterBot is a library in python which generates responses to user input.
- ✓ It uses a number of machine learning algorithms to produce a variety of responses.
- ✓ It becomes easier for the users to make chatbots using the ChatterBot library with more accurate responses.



Purpose of chatbot :

- ✓ Rule-based customer service chatbots are often used for straightforward tasks such as providing basic information, answering frequently asked questions or performing simple transactions.
- ✓ Rule-based bots give you more control over how your chatbots operate at every stage of the interaction.

Python library:

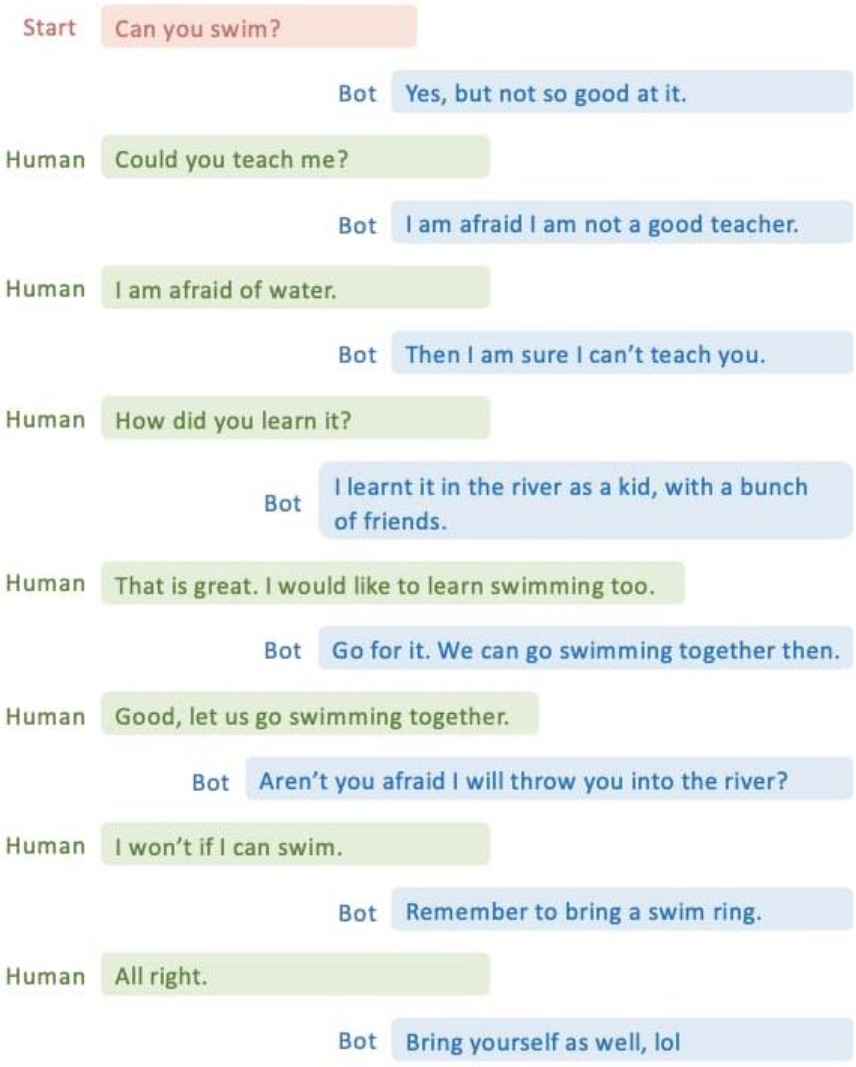
Chatterbot:

- ✓ ChatterBot is a Python library designed to make it easy to create software that can engage in conversation.
- ✓ It uses a selection of machine learning algorithms to produce different types of responses.
- ✓ This helps create chatbots and automate conversations with users.

Dataset link:

https://www.kaggle.com/datasets/grafstor/simple-dialogs-for-chatbot

Sample dataset :



Chatbot logic:

Chatbot Logic: Understand & Generate

Chatbot logic refers to the underlying framework and algorithms that enable a chatbot to understand and generate human-like text in a conversation.

It involves natural language processing (NLP) techniques, machine learning, and various components such as:

Input Processing: The chatbot processes user inputs to extract intent, entities, and context. This is often done through techniques like tokenization and part-of-speech tagging.

Intent Recognition: Determining the user's intention or purpose of the message, such as asking a question, making a request, or providing information.

Entity Recognition: Identifying specific pieces of information in the user's message, like names, dates, or locations.

Context Management: Keeping track of the conversation history to understand and generate responses that are contextually relevant.

Response Generation: Crafting responses based on the recognized intent, entities, and context. This can be done using rule-based systems, machine learning models, or a combination of both.

Machine Learning Models: Many chatbots use machine learning models like recurrent neural networks or transformer models to generate text responses.

Dialog Flow: Managing the flow of the conversation, ensuring that responses are coherent and relevant to the ongoing discussion.

User Experience: Consideration of user experience principles to make the interaction with the chatbot more natural and user-friendly.

Testing and Iteration: Continuously improving the chatbot by analyzing user interactions, identifying areas for improvement, and fine-tuning the logic and responses.

Chatbot logic can be as simple as rule-based systems or as complex as advanced machine learning models, depending on the chatbot's purpose and capabilities.

Design user interface:

Designing a user interface for a chatbot involves several key elements:

Chat Window: A central chat window where the user can type messages and receive responses.

User Input Field: A text input field at the bottom for users to type their messages.

Send Button: A button near the input field to send messages.

Chat Bubbles: Messages from the chatbot and user should be displayed in chat bubbles, typically on alternating sides.

Profile Images: Small profile images or icons to represent the user and the chatbot.

Typing Indicator: Display a typing indicator when the chatbot is responding.

Menu/Commands: A menu or list of available commands if the chatbot has specific functions.

Attachment Options: Allow users to attach files or media if necessary.

Emojis/GIFs: Optionally, provide access to emojis or GIFs for a more engaging conversation.

Help/FAQ: Include a button or option to access frequently asked questions or help.

Minimize/Maximize: Allow users to minimize or maximize the chat window.

Scrollable History: Ensure that the chat history is scrollable to view past messages.

Responsive Design: Make sure the interface is responsive for various screen sizes.

Customization Options: Provide options for users to customize the chatbot's appearance or behavior, if applicable.

Feedback Option: Include a way for users to provide feedback or report issues.

Sample program:

```
import warnings

warnings.filterwarnings('ignore')

import numpy as np

import pandas as pd
```

```
import matplotlib.pyplot as plt
import tensorflow as tf
import keras
from tqdm import tqdm
from keras.layers import Dense
import json
import re
import string
from sklearn.feature_extraction.text import TfidfVectorizer
import unicodedata
from sklearn.model_selection import train_test_split
question  =[]
answer = []
with open("../input/simple-dialogs-for-chatbot/dialogs.txt",'r') as f :
    for line in f :
        line  =  line.split('\t')
        question.append(line[0])
        answer.append(line[1])
print(len(question) == len(answer))
```

```
True
```

In [3]:

```
question[:5]
```

Out[3]:

```
['hi, how are you doing?',
 "i'm fine. how about yourself?",
 "i'm pretty good. thanks for askin
g.",
 'no problem. so how have you been?',
 "i've been great. what about you?"]
```

In [4]:

```
answer[:5]
```

Out[4]:

```
["i'm fine. how about yourself?\n",
 "i'm pretty good. thanks for askin
g.\n",
 'no problem. so how have you been?
\n',
 "i've been great. what about you?
\n",
 "i've been good. i'm in school right
now.\n"]
```

In [5]:

```
answer = [ i.replace("\n","") for i i
n answer]
```

In [6]:

```
answer[:5]
```

Out[6]:

```
["i'm fine. how about yourself?",
 "i'm pretty good. thanks for askin
g.",
 'no problem. so how have you been?',
 "i've been great. what about you?",
 "i've been good. i'm in school right
now."]
```

```
data = pd.DataFrame({"question" : question ,"answer":answer})
data.head()
```

Output:

	question	answer
0	hi, how are you doing?	i'm fine. how about yourself?
1	i'm fine. how about yourself?	i'm pretty good. thanks for asking.
2	i'm pretty good. thanks for asking.	no problem. so how have you been?
3	no problem. so how have you been?	i've been great. what about you?
4	i've been great. what about you?	i've been good. i'm in school right now.

Conclusion:

The sample program for the chatbot and sample output for the chatbot is added.The chatbot is developed furtherly innovated in more steps and ways that are been developed with the questions and answers.

>>>THANK YOU<<<

Presented by:-

Ahila. K

Bhuvaneshwari. S

Gopika. K

Harshini. A

Kavya. K