CHAT BOT

ABSTRACT:

A chatbot enables a user to simply ask questions in the same manner that they would respond to humans. The most well-known chatbots currently are voices chatbots: SIRI and Alexa. However, chatbots have been adopted and brought into the daily application at a high rate on the computer chat platform. NLP also allows computers and algorithms to understand human interactions through various languages. Recent advances in machine learning have greatly improved the accurate and effective of natural language processing, making chatbots a viable option for many organizations. This improvement in NLP is firing a great deal of additional research which should lead to continued improvement in the effective of chatbots in the years to come.A bot is trained on and according to the training, based on some rules on which it is trained, it answers questions. It is called ruled based approach. The language by which these bots can be created is Artificial Intelligence Markup Language (AIML). It is a language based on XML which allows the developer to write the rules which the bot will follow. In this research paper, We are trying to understand these chatbots and understanding their shortcomings. question or statement submitted by a user and allow the user to control over the content to be displayed

OBJECTIVE:

Imagine having a ride all the way to an incorrect destination. It would be a waste of time and effort. The same goes for building your chatbot. If you are not clear with the problem statement/ use-case, you might get lost and end up with a fancy but impractical chatbot.

The right bot for your business lies at the intersection of your objective(customer support, lead generation, recommendation), the

right technology (rule-based, AI chatbot, or hybrid), and the channel to build it in (slack, Facebook messenger, social media, website, mobile).

INTRODUCTION:

The improvements in the fields of inter-networking and information technology have been intricate in executing an Artificial Intelligent systems. These systems are drawing near to human activities for example choice emotionally supportive networks, robotics, natural language processing. Indeed, even in the artificial intelligent fields there are some hybrid strategies and adaptive techniques that make increase complex techniques. That, yet these days there are additionally several Natural Language Processing and intelligent systems that could comprehend human language. AI systems learn themselves and retrieve insight by perusing required electronic articles that have been exist on the web page.

A chatbot is an AI program that copy human discussions including content and communication in natural language utilizing artificial intelligence method for example, Natural Language Processing is a picture and video processing and voice analysis. chatbot for college management system has been created utilizing AI algorithms that examine the user queries. This chatbot system is an internet

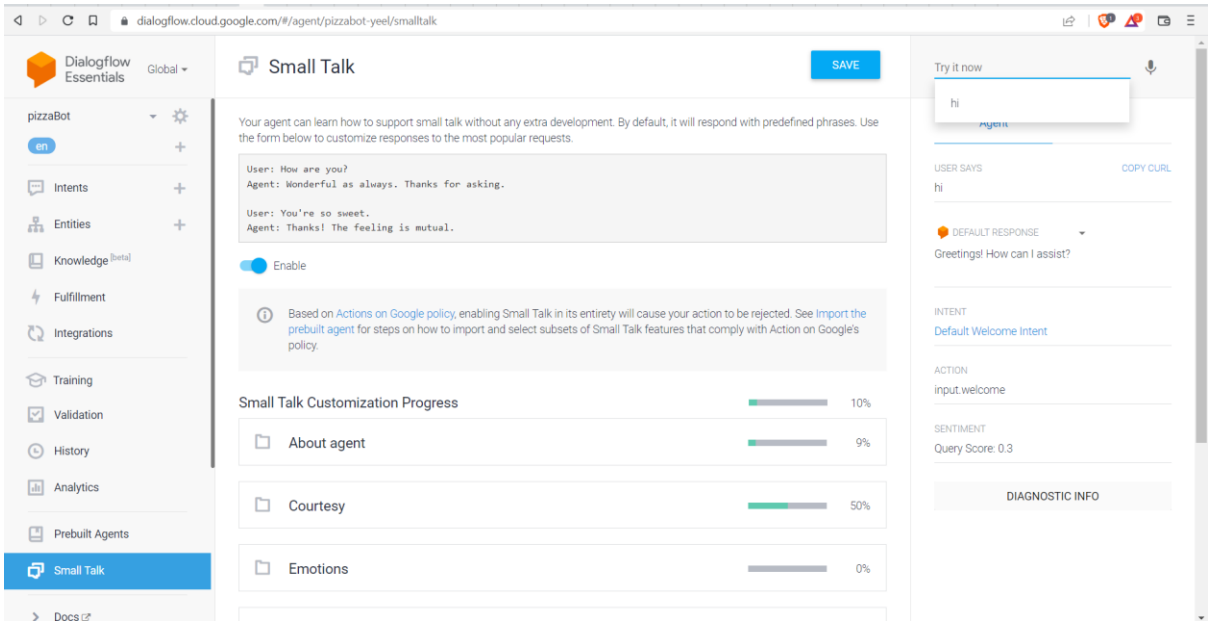
application that gives an answer to the broken down queries of a user. Users simply need to choose the classification for inquiries and afterward they can ask the question to the bot that utilizes for noting it. AI has been incorporated to respond to the users inquiries then the user can procure the fitting solutions to their inquiries.

Chatbot has become the centre of focus in this current era thus the bot are being utilized to deliver information more conveniently. A chatbot is a standout amongst the most progressive and promising tools of communication among people and machines. famous chatbots like amazon Alexa, Siri, Facebook, Slack, and many more are in trend. These are very much helpful but in this era of enhancing technology day by day, technology gets updated and accordingly by the user expectations also increases. A user wants more automation in the chatbot although every system is not perfect but there is always a flaw in the system. so in the chatbot there are some problems that the user has experienced while using a chatbot. A chatbot can be described as an answering system where a system will be able to answer questions or to the statement submitted by users and allow users to control over the content to be displayed

DIALOG FLOW:

Dialog flow is a key factor for chatbots. You can create a logical dialog flow based on the type of questions encountered by the bot. It should be a detailed response, which requires defining the information for each response. Every dialog flow design should contain the exact representations of the response on each question. The design for

detailed answers should happen outside of the actual flow design, as you may want to have variants of the same answers based on the questions. This is known as Random Prompting and it is a technique that you use while developing chatbots.



METHODOLOGY:

Artificial Intelligence is the driving force behind the creation of innovative products like autonomous vehicles and chatbots. Recent advancements in Natural Language Processing (NLP) have made chatbots, also referred to as virtual assistants, a great option for improving the customer experience. Answering frequently asked questions, filing claims, checking the status of an order and getting feedback from customers are among the most popular use cases for chatbots. **Building a chatbot that offers a good experience to customers** requires collaboration from an interdisciplinary team of business analysts, service designers, data scientists, machine learning engineers and software developers. The chatbot development methodology blends several modern frameworks and methodologies including design thinking, AI innovation sprints, and agile software development.

CODE:

```
# -*- coding: utf-8 -*-
"""Untitled1.ipynb

Automatically generated by Colaboratory.

Original file is located at
    https://colab.research.google.com/drive/18SC6QpqPoq0BvG_J5BfWuCijPMEF7zLs
"""

import numpy as np
import nltk
import string
import random

from nltk.corpus.reader.tagged import word_tokenize
f=open("/content/human_chat.txt",'r',errors='ignore')
raw_doc=f.read()
raw_doc=raw_doc.lower()
nltk.download('punkt')
nltk.download('wordnet')
sent_tokens=nltk.sent_tokenize(raw_doc)
word_tokens=nltk.word_tokenize(raw_doc)

sent_tokens[:2]

word_tokens[:2]

lemmer=nltk.stem.WordNetLemmatizer()
def LemTokens(tokens):
  return[lemmer.lemmatize(token) for token in tokens]
remove_punct_dict=dict((ord(punct),None)for punct in string.punctuation)
def LemNormalize(text):
  return
LemTokens(nltk.word_tokenize(text.lower().translate(remove_punct_dict)))

GREET_INPUTS=("hello","hi","greeting","sup","what's up","hey",)
GREET_RESPONSES=["hi","hey","*nods*","hi there","hello","I am glad You are
talking to me"]
def greet(sentence):
  for word in sentence.split():
    if word.lower() in GREET_INPUTS:
      return random.choice(GREET_RESPONSES)

from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity
```

```
def response(user_response):
  robo1_response=''
  TfidfVec=TfidfVectorizer(tokenizer=LemNormalize,stop_words='english')
  tfidf=TfidfVec.fit_transform(sent_tokens)
  vals=cosine_similarity(tfidf[-1],tfidf)
  idx=vals.argsort()[0][-2]
  flat=vals.flatten()
  flat.sort()
  req_tfidf=flat[-2]
  if(req_tfidf==0):
    robo1_response=robo1_response+"I am sorry! I don't understand you"
    return robo1_response
  else:
    robo1_response=robo1_response+sent_tokens[idx]
    return robo1_response

flag=True
print("Bot:My name is hanoch.Let's have a converstion!Also,if you want to exit
any time,just type Bye")
while(flag==True):
  user_response=input()
  user_response=user_response.lower()
  if(user_response!='bye'):
    if(user_response=='thanks'or user_response=='thank you'):
      flag=False
      print("Bot:You are welcome..")
    else:
      if(greet(user_response)!=None):
        print("Bot:"+greet(user_response))
      else:
         sent_tokens.append(user_response)
         word_tokens=word_tokens+nltk.word_tokenize(user_response)
         final_words=list(set(word_tokens))
         print("Bot:",end="")
         print(response(user_response))
         sent_tokens.remove(user_response)
  else:
    flag=False
    print("Bot:Goodbye! Take care <3")
```

+ Code + Text

Busy | Editing

```
import numpy as np
import nltk
import string
import random
```

```
from nltk.corpus.reader.tagged import word_tokenize
f=open("/content/human_chat.txt",'r',errors='ignore')
raw_doc=f.read()
raw_doc=raw_doc.lower()
nltk.download('punkt')
nltk.download('wordnet')
sent_tokens=nltk.sent_tokenize(raw_doc)
word_tokens=nltk.word_tokenize(raw_doc)
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
[nltk_data] Downloading package wordnet to /root/nltk_data...
```

```
sent_tokens[:2]
```

```
['human 1: hi!', 'human 2: what is your favorite holiday?']
```

```
word_tokens[:2]
```

```
['human', '1']
```

```
lemmer=nltk.stem.WordNetLemmatizer()
def LemTokens(tokens):
```

4m 47s completed at 9:19 PM

+ Code + Text

Busy | Editing

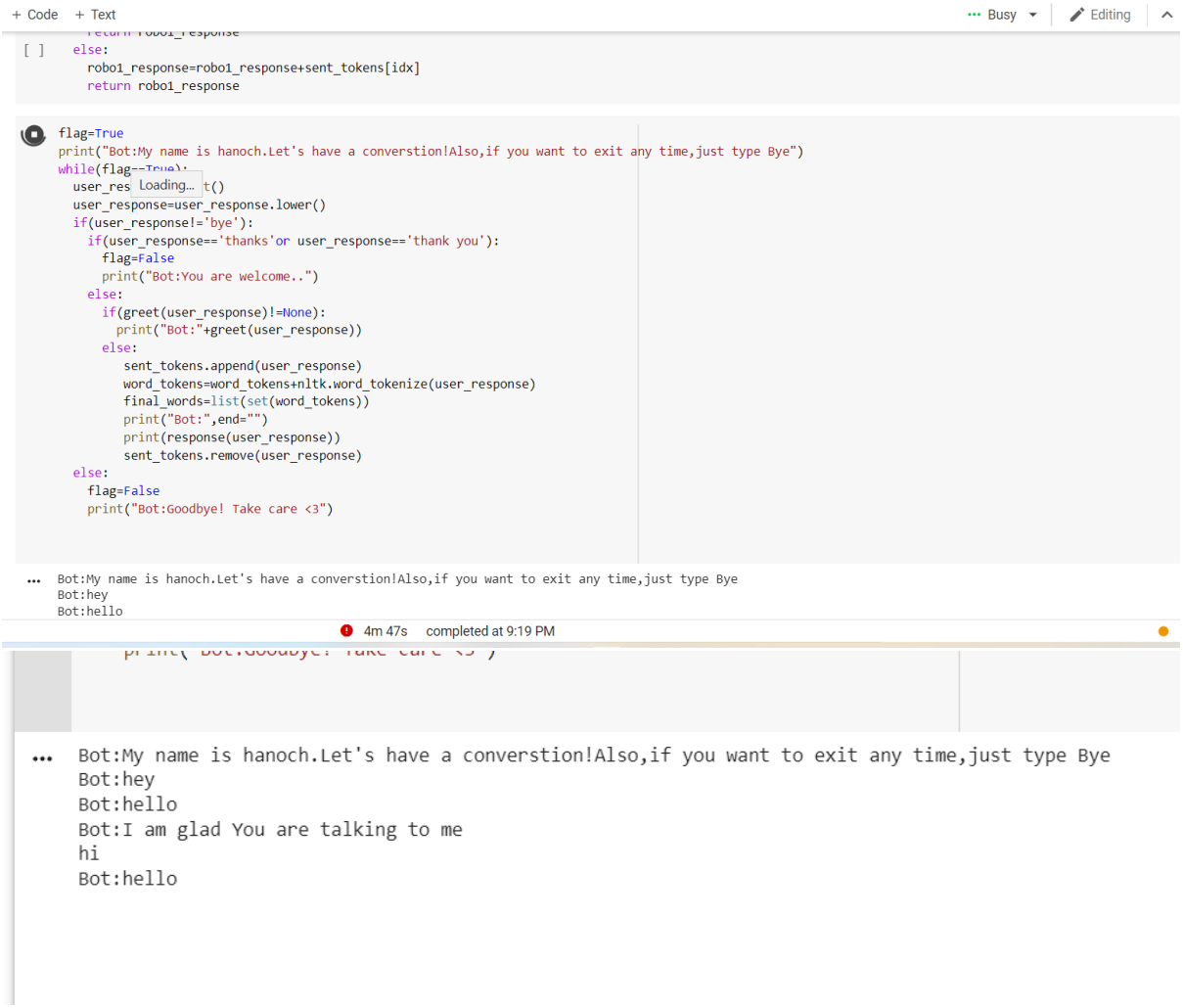
```
lemmer=nltk.stem.WordNetLemmatizer()
def LemTokens(tokens):
  return[lemmer.lemmatize(token) for token in tokens]
remove_punct_dict=dict((ord(punct),None)for punct in string.punctuation)
def LemNormalize(text):
  return LemTokens(nltk.word_tokenize(text.lower().translate(remove_punct_dict)))
```

```
GREET_INPUTS=("hello","hi","greeting","sup","what's up","hey",)
GREET_RESPONSES=["hi","hey","*nods*","hi there","hello","I am glad You are talking to me"]
def greet(sentence):
  for word in sentence.split():
    if word.lower() in GREET_INPUTS:
      return random.choice(GREET_RESPONSES)
```

```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity
```

```
def response(user_response):
  robo1_response=''
  TfidfVec=TfidfVectorizer(tokenizer=LemNormalize,stop_words='english')
  tfidf=TfidfVec.fit_transform(sent_tokens)
  vals=cosine_similarity(tfidf[-1],tfidf)
  idx=vals.argsort()[0][-2]
  flat=vals.flatten()
  flat.sort()
  req_tfidf=flat[-2]
  if(req_tfidf==0):
    robo1_response=robo1_response+"I am sorry! I don't understand you"
    return robo1_response
  else:
```

4m 47s completed at 9:19 PM



CONCLUSION

In this project, we made a college-specific chatbot system that can be custom and fits in an education domain chatbot the addition of this chatbot system in the college website will make the webpage more user interactive as it responds to the user queries very accurately as it is a domain-specific chatbot system, and furthermore we had investigated our college chatbot system design stages. a few different techniques by which the precision of the chatbot system can be made better. gathering feedback from the potential user can be helpful in developing the college Chatbot system ultimately servicing the user queries in conclusion we have made a chatbot in python that can understand user queries and reply accordingly. In the intent file of our chatbot on we can add more patterns and improve patterns which will be helpful when replying to the users and improve the accuracy of our chatbot DL enabled chatbots are becoming more and more popular because of their applications and they can tackle all the problem. it can

also be very helpful in teaching and has a lot of applications in teaching the visually impaired.