**Implementation**

Pip install nltk

Import nltk

Import io

Import numpy as np

Import random

Import string to process standard python strings

Import warnings

Warnings.filterwarnings(‘ignore’)

Opening file and reading:-

f=open('D:\globalwarming.txt','r',errors= 'ignore')

raw=f.read()

raw=raw.lower()# converts to Lowercase

nltk.download('punkt') # first-time use only

nltk.download('wordnet') # first-time use only

sent\_tokens = nltk.sent\_tokenize(raw) # converts to list of sentences

word\_tokens = nltk.word\_tokenize(raw) # converts to list of words

**punktSentence Tokenizer**

**This tokenizer divides a text into a list of sentence**

**WordNet is a lexical database for the English Language, which has created by Princeton and is part of the NLTK corpus.**

sent\_tokens[:2]

word\_tokens[:2]

Pre-processing

lemmer = nltk.stem.WordNetLemmatizer()

#WordNet is a semantically-oriented dictionary of English included in NLTK.

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)))

Greeting

GREETING\_INPUTS= ("hello", "hi", "greetings", "sup", "what's up", "hey",)

GREETING RESPONSES = ["hi", "hey", "\*nods\*", "hi there", "hello", "I am glad! You are talking to me"]

def greeting(sentence):

for word in sentence.split():

if word.lower() in GREETING\_INPUTS:

return random.choice (GREETING\_RESPONSES)

Vectorizer

from sklearn.feature\_extraction.text import TfidfVectorizer

from sklearn.metrics.pairwise import cosine\_similarity

def response(user\_response):

chatbot\_response=''

sent\_tokens.append(user\_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):

chatbot\_response-chatbot\_response+"I am sorry! I don't understand you"

return chatbot\_response

else:

chatbot\_response = chatbot\_response+sent\_tokens[idx]

return chatbot\_response

flag=True

print("Chatbot: My name is Chatbot. I will answer your queries about Global Warming. If you want to exit, 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("Chatbot: You are welcome..")

else:

if (greeting(user\_response)!=None):

print("Chatbot: "+greeting (user\_response))

else:

print("Chatbot: ",end="")

print (response(user\_response))

sent\_tokens.remove(user\_response)

else:

flag=False

print("Chatbot: Bye! take care..")