Chat-Bot using NLTK

Project based on Python in the sphere of AI used to provide customer assistance.

Importing the required libraries

In [1]:

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

Importing and reading the corpus

In [2]:

```
f=open('D:\Ankit\Database\chatbot.txt','r',errors = 'ignore')
raw_doc=f.read()
raw_doc=raw_doc.lower() #Converts text to lowercase
nltk.download('punkt') #Using the Punkt tokenizer
nltk.download('wordnet') #Using the WordNet dictionary
sent_tokens = nltk.sent_tokenize(raw_doc) #Converts doc to list of sentences
word_tokens = nltk.word_tokenize(raw_doc) #Converts doc to list of words
```

```
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\Ankit\AppData\Roaming\nltk_data...
[nltk_data]   Unzipping tokenizers\punkt.zip.
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\Ankit\AppData\Roaming\nltk_data...
[nltk_data]   Unzipping corpora\wordnet.zip.
```

Example of sentance tokens

In [3]:

```
sent_tokens[:2]
```

Out[3]:

```
['data science is an interdisciplinary field that uses scientific methods, processe
s, algorithms and systems to extract knowledge and insights from noisy, structured a
nd unstructured data,[1][2] and apply knowledge and actionable insights from data ac
ross a broad range of application domains.',
 'data science is related to data mining, machine learning and big data.']
```

Example of word tokens

In [4]:

```
word_tokens[:2]
```

Out[4]:

```
['data', 'science']
```

Text preprocessing

In [5]:

```
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)))
```

Defining the greeting function

In [6]:

```
GREET_INPUTS = ("hello", "hi", "greetings", "sup", "what's up","hey")
GREET_RESPONSES = ["hi", "hey", "*nods*", "hi there", "hello", "I am glad! You are t
def greet(sentence):
    for word in sentence.split():
        if word.lower() in GREET_INPUTS:
            return random.choice(GREET_RESPONSES)
```

Response generation

In [7]:

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

In [8]:

```
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
```

Defining conversation start/end protocols

In [9]:

```
flag=True
print("BOT: My name is Stark. Let's have a conversation! Also, if you want to exit a
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 ")
```

```
BOT: My name is Stark. Let's have a conversation! Also, if you want to exit any tim
e, just type Bye!
Hi
BOT: hey
```

```
technology
BOT: 
```

```
C:\Users\Ankit\anaconda4\lib\site-packages\sklearn\feature_extraction\text.py:388: U
serWarning: Your stop_words may be inconsistent with your preprocessing. Tokenizing
the stop words generated tokens ['ha', 'le', 'u', 'wa'] not in stop_words.
  warnings.warn('Your stop_words may be inconsistent with '
```

```
[30]

technologies and techniques
there is a variety of different technologies and techniques that are used for data s
cience which depend on the application.
Foundations
BOT: [6]


contents
1       foundations
1.1     relationship to statistics
2       etymology
2.1     early usage
2.2     modern usage
3       technologies and techniques
4       see also
5       references
foundations
data science is an interdisciplinary field focused on extracting knowledge from data
sets, which are typically large (see big data), and applying the knowledge and actio
nable insights from data to solve problems in a wide range of application domains.
bye
BOT: Goodbye! Take care <3
```