Create a chat bot using python

Chat Bot in Python with Chatter Bot Module

Nobody likes to be alone always, but sometimes loneliness could be a better medicine to hunch the thirst for a peaceful environment. Even during such lonely quarantines, we may ignore humans but not humanoids. Yes, if you have guessed this article for a chatbot, then you have cracked it right. We won't require 6000 lines of code to create a chatbot but just a six-letter word "Python" is enough. Let us have a quick glance at Python's **ChatterBot** to create our bot. ChatterBot is a Python library built based on machine learning with an inbuilt conversational dialog flow and training engine. The bot created using this library will get trained automatically with the response it gets from the user.

Understanding the Chatbot

A **Chatbot** is an Artificial Intelligence-based software developed to interact with humans in their natural languages. These chatbots are generally converse through auditory or textual methods, and they can effortlessly mimic human languages to communicate with human beings in a human-like way. A chatbot is considered one of the best applications of natural languages processing.

We can categorize the Chatbots into two primary variants: **Rule-Based Chatbots** and **Self-Learning Chatbots**.

1. **Rule-based Chatbots:**The Rule-based approach trains a chatbot to answer questions based on a list of pre-determined rules on which it was primarily trained. These set rules can either be pretty simple or quite complex, and we can use these rule-based chatbots to handle simple queries but not process more complicated requests or queries.

2. **Self-learning Chatbots:**Self-learning chatbots are chatbots that can learn on their own. These leverage advanced technologies such as Artificial Intelligence (AI) and Machine Learning (ML) to train themselves from behaviours and instances. Generally, these chatbots are quite smarter than rule-based bots. We can classify the Self-learning chatbots furtherly into two categories - **Retrieval-based Chatbots** and **Generative Chatbots**.
    a. **Retrieval-based Chatbots:**A retrieval-based chatbot works on pre-defined input patterns and sets responses. Once the question or pattern is inserted, the chatbot utilizes a heuristic approach to deliver the relevant response. The model based on retrieval is extensively utilized to design and develop goal-oriented chatbots using customized features such as the flow and tone of the bot in order to enhance the experience of the customer.
    b. **Generative Chatbots:**Unlike retrieval-based chatbots, generative chatbots are not based on pre-defined responses - they leverage seq2seq neural networks. This is constructed on the concept of machine translation, where the source code is converted from one language to another language. In the seq2seq approach, the input is changed into an output.

The first chatbot named **ELIZA** was designed and developed by Joseph Weizenbaum in 1966 that could imitate the language of a psychotherapist in only 200 lines of code. But as the technology gets more advance, we have come a long way from scripted chatbots to chatbots in Python today.

Chatbot in present Generation

Today, we have smart Chatbots powered by Artificial Intelligence that utilize natural language processing (NLP) in order to understand the commands from humans (text and voice) and learn from experience. Chatbots have become a staple customer interaction utility for companies and brands that have an active online existence (website and social network platforms).

With the help of Python, Chatbots are considered a nifty utility as they facilitate rapid messaging between the brand and the customer. Let us think about Microsoft's Cortana, Amazon's Alexa, and Apple's Siri. Aren't these chatbots wonderful? It becomes quite interesting to learn how to create a chatbot using the Python programming language.

Fundamentally, the chatbot utilizing Python is designed and programmed to take in the data we provide and then analyze it using the complex algorithms for Artificial Intelligence. It then delivers us either a written response or a verbal one. Since these bots can learn from experiences and behavior, they can respond to a large variety of queries and commands.

Although chatbot in Python has already started to rule the tech scenario at present, chatbots had handled approximately 85% of the customer-brand interactions by 2020 as per the prediction of Gartner.

In light of the increasing popularity and adoption of chatbots in the industry, we can increase the market value by learning how to create a chatbot in Python - among the most extensively utilized programming languages globally.

Understanding the ChatterBot Library

ChatterBot is a Python library that is developed to provide automated responses to user inputs. It makes utilization of a combination of Machine Learning algorithms in order to generate multiple types of responses. This feature enables developers to construct chatbots using Python that can communicate with humans and provide relevant and appropriate responses. Moreover, the ML algorithms support the bot to improve its performance with experience.

Another amazing feature of the **ChatterBot** library is its language independence. The library is developed in such a manner that makes it possible to train the bot in more than one programming language.

Why Chatbots are important for a Business or a Website

- Quick resolution for a complaint or a problem.
- Improve business branding thereby achieving great customer satisfaction.
- Answering questions and answers for customers.
- Making a reservation at hotel or at restaurant.
- Save human effort 24×7.
- Enhance business revenue by providing ideas and inspirations.
- Finding details about business such as hours of operation, phone number and address.
- Automate sales and lead generation process.
- Reduce customer agents waiting time answering phone calls.

Benefits of using Chatbots

- 24×7 availability.
- Instant answers to queries.
- Support multi-language to enhance businesses.
- Simple and Easy to Use UI to engage more customers.
- Cost effective and user interactive.
- Avoid communication with call agents thereby reducing the time consuming tasks.
- Understand the Customer behavior
- Increase sales of business by offering promo codes or gifts.

Types of Chatbots

Chatbots deliver instantly by understanding the user requests with pre-defined rules and AI based chatbots. There are two types of chatbots.

- **Rule Based Chatbots:** This type of chatbots answer the customer queries using the pre-defined rules. These bots answer common queries such as hours of operation of business, addresses, phone numbers and tracking status.

- **Conversational AI Chatbots:** This type of chatbots using Natural language Processing(NLP) to understand the context and intent of a user input before providing the response. These Bots train themselves as per the user inputs and more they learn, more they become user interactive.

Installation

Install chatterbot using Python Package Index(PyPi) with this command

```
pip install chatterbot
```

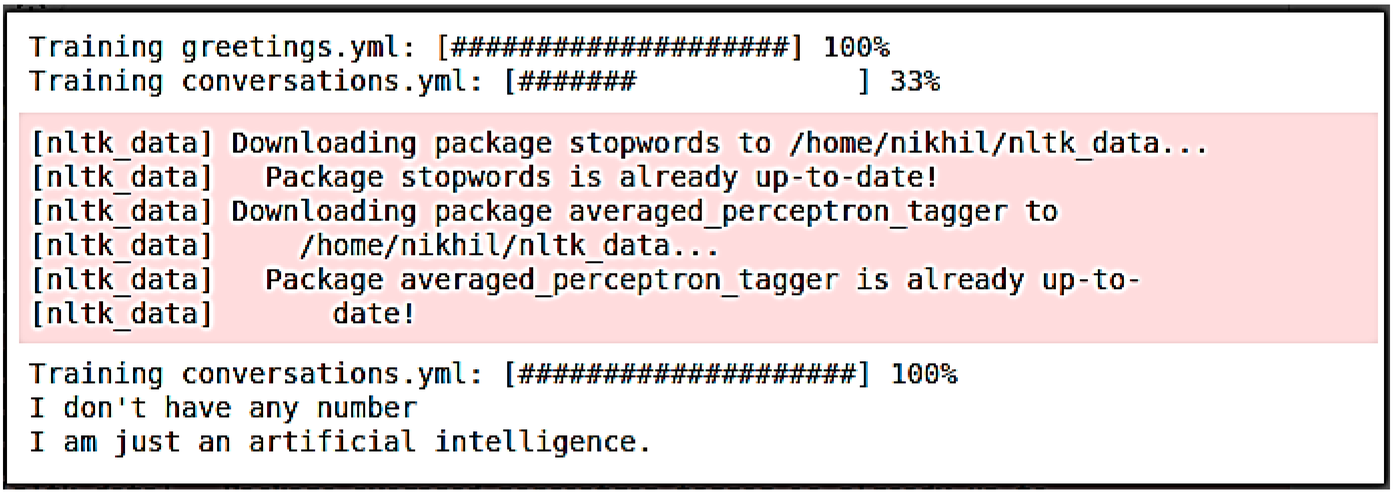
Below is the implementation.

Example program

```
# Import "chatbot" from

# chatterbot package.

from chatterbot import ChatBot

# Inorder to train our bot, we have

# to import a trainer package

# "ChatterBotCorpusTrainer"

from chatterbot.trainers import ChatterBotCorpusTrainer

# Give a name to the chatbot “corona bot”

# and assign a trainer component.
```

```
chatbot=ChatBot('corona bot')

# Create a new trainer for the chatbot

trainer = ChatterBotCorpusTrainer(chatbot)

# Now let us train our bot with multiple corpus

trainer.train("chatterbot.corpus.english.greetings",

                    "chatterbot.corpus.english.conversations" )

response = chatbot.get_response('What is your Number')

print(response)

response = chatbot.get_response('Who are you?')

print(response)
```

output



```
Training greetings.yml: [####################] 100%
Training conversations.yml: [#######             ] 33%

[nltk_data] Downloading package stopwords to /home/nikhil/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     /home/nikhil/nltk_data...
[nltk_data]   Package averaged_perceptron_tagger is already up-to-
[nltk_data]       date!

Training conversations.yml: [####################] 100%
I don't have any number
I am just an artificial intelligence.
```

Report:

All the above instruction are installed and execution successfully

<u>Project by:</u>

Name: N.Ramesh

Dept: 3rd year CSE

Reg no: 621421104041

College code:6214

Group: IBM-Group 5