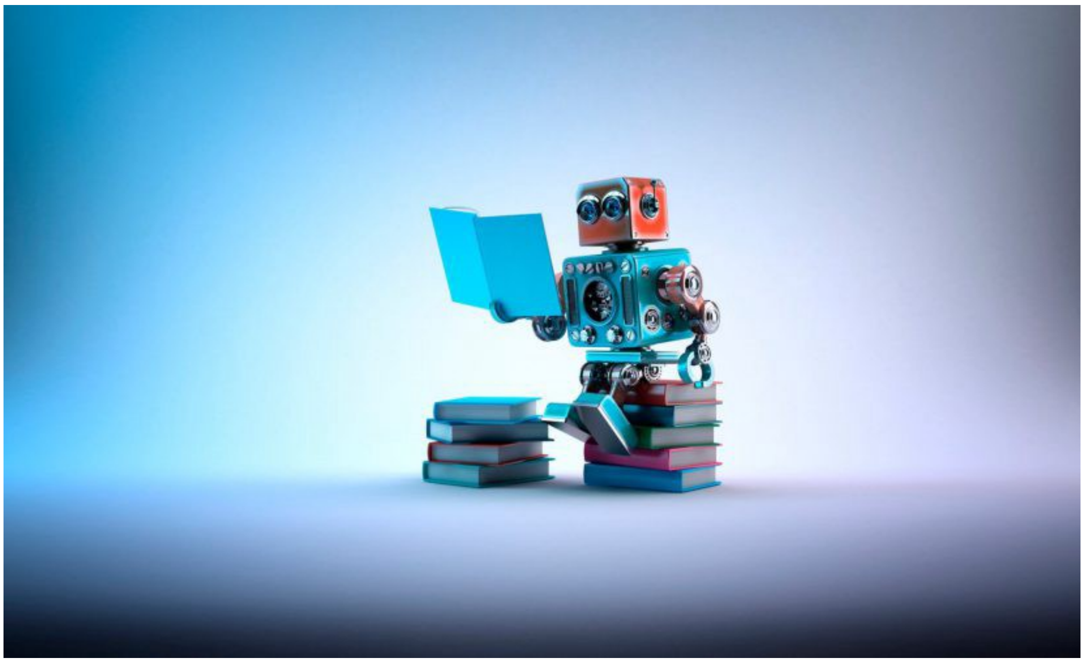
Build your first chatbot using Python NLTK

Nagesh Singh
Chauhan
Oct 17, 2018 · 5 min read


source : https://octodev.net/most-advanced-chatbot-apps-powered-by-artificial-intellect/

"A **chatbot** (also known as a **talkbot**, **chatterbot**, **Bot**, **IM bot**, **interactive agent**, or **Artificial Conversational Entity**) is a computer program or an artificial intelligence which conducts a conversation via auditory or textual methods. Such programs are often designed to convincingly simulate how a human would behave as a conversational partner, thereby passing the Turing test. Chatbots are typically used in dialog systems for various practical purposes including customer service or information acquisition. Some chatterbots use sophisticated natural language processing systems, but many simpler systems scan for keywords within the input, then pull a reply with the most matching keywords, or the most similar wording pattern, from a database."

— source wikipedia

Chatbots are not very new, one of the foremost of this kind is ELIZA, which was created in the early 1960s and is worth exploring. In order to successfully build a conversational engine, it should take care of the following things:

1. Understand who is the target audience
2. Understand the Natural Language of the communication.
3. Understand the intent or desire of the user
4. provide responses that can answer the user

Today we will learn to create a simple chat assistant or chatbot using Python's NLTK

Today we will learn to create a simple chat assistant or chatbot using Python's NLTK library.

NLTK has a module, nltk.chat, which simplifies building these engines by providing a generic framework.

In this blog I am using 2 imports from nltk.chat.util:

Chat: This is a class that has all the logic that is used by the chatbot.

Reflections: This is a dictionary that contains a set of input values and its corresponding output values. It is an optional dictionary that you can use. You can also create your own dictionary in the same format as below and use it in your code. If you check nltk.chat.util, you will see its values as below:

```
reflections = {
  "i am"       : "you are",
  "i was"      : "you were",
  "i"          : "you",
  "i'm"        : "you are",
  "i'd"        : "you would",
  "i've"       : "you have",
  "i'll"       : "you will",
  "my"         : "your",
  "you are"    : "I am",
  "you were"   : "I was",
  "you've"     : "I have",
  "you'll"     : "I will",
  "your"       : "my",
  "yours"      : "mine",
  "you"        : "me",
  "me"         : "you"
}
```

You can also create your own reflections dictionary in the same format as above and use it in your code. Here is an example for this:

```
my_dummy_reflections= {
    "go"     : "gone",
    "hello"    : "hey there"
}
```

and use it as :

```
chat = Chat(pairs, my_dummy_reflections)
```

Using above concept from python's NLTK library, lets build a simple chatbot without using any of the Machine Learning or Deep Learning Algorithms. So obviously our chatbot will be a decent one but not an intelligent one.

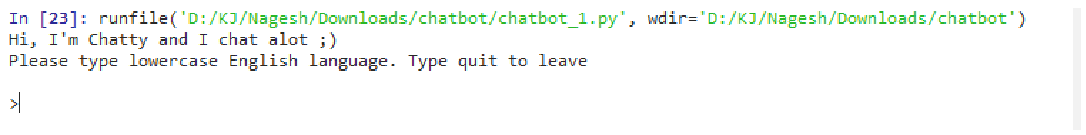
Source Code :

```
from nltk.chat.util import Chat, reflections

pairs = [
    [
        r"my name is (.*)",
        ["Hello %1, How are you today ?",]
    ],
     [
        r"what is your name ?",
        ["My name is Chatty and I'm a chatbot ?",]
    ],
    [
        r"how are you ?",
        ["I'm doing good\nHow about You ?",]
    ],
    [
        r"sorry (.*)",
        ["Its alright","Its OK, never mind",]
    ],
    [
        r"i'm (.*) doing good",
        ["Nice to hear that","Alright :)",]
    ],
    [
        r"hi|hey|hello",
        ["Hello", "Hey there",]
    ],
    [
        r"(.*) age?",
        ["I'm a computer program dude\nSeriously you are asking me this?",]

    ],
    [
        r"what (.*) want ?",
        ["Make me an offer I can't refuse",]

    ],
    [
        r"(.*) created ?",
        ["Nagesh created me using Python's NLTK library ","top secret ;)",]
    ],
    [
        r"(.*) (location|city) ?",
        ['Chennai, Tamil Nadu',]
    ],
    [
        r"how is weather in (.*)?",
        ["Weather in %1 is awesome like always","Too hot man here in %1","Too cold man here in %1","Never
even heard about %1"]
    ],
    [
        r"i work in (.*)?",
        ["%1 is an Amazing company, I have heard about it. But they are in huge loss these days.",]
    ],

[
        r"(.*)raining in (.*)",
        ["No rain since last week here in %2","Damn its raining too much here in %2"]
    ],
    [
        r"how (.*) health(.*)",
        ["I'm a computer program, so I'm always healthy ",]
    ],
    [
        r"(.*) (sports|game) ?",
        ["I'm a very big fan of Football",]
    ],
    [
        r"who (.*) sportsperson ?",
        ["Messy","Ronaldo","Roony"]
```

```
],
    [
        r"who (.*) (moviestar|actor)?",
        ["Brad Pitt"]

],
    [
        r"quit",
        ["BBye take care. See you soon :) ","It was nice talking to you. See you soon :)"]

],
]
def chatty():
        print("Hi, I'm Chatty and I chat alot ;)\nPlease type lowercase English language to start a
conversation. Type quit to leave ") #default message at the start

    chat = Chat(pairs, reflections)
    chat.converse()
if __name__ == "__main__":
    chatty()
```

The code is quite simple, still lets understand it.

Once the function chatty() is invoked , a default message will be displayed:



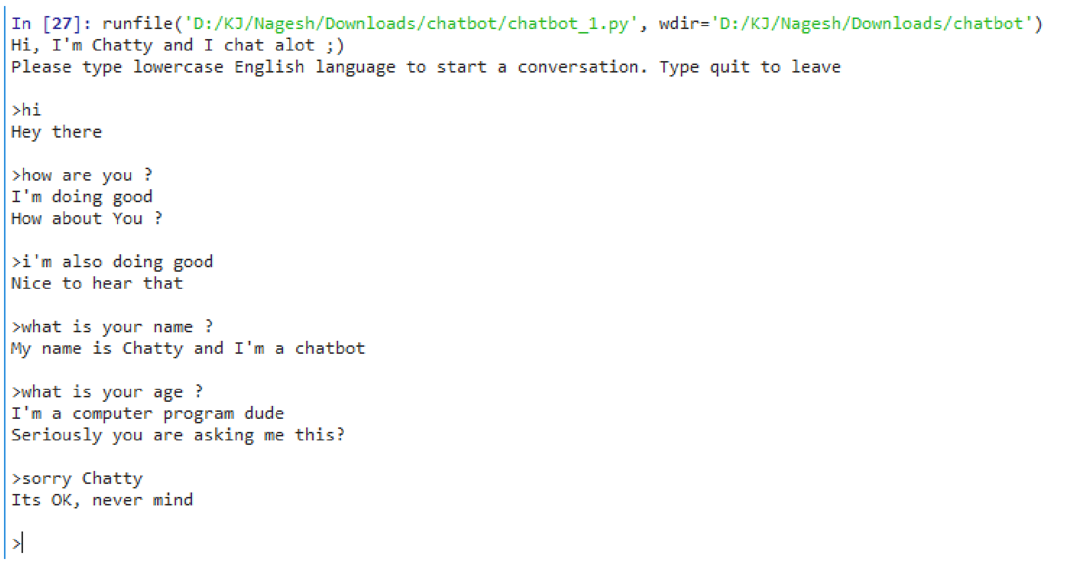
```
In [23]: runfile('D:/KJ/Nagesh/Downloads/chatbot/chatbot_1.py', wdir='D:/KJ/Nagesh/Downloads/chatbot')
Hi, I'm Chatty and I chat alot ;)
Please type lowercase English language. Type quit to leave

>
```

Next I’ve created an instance of Chat class containing **pairs**(list of tuples containing set of question and answers) and **reflections**(discussed above).

Next step is to trigger the conversation:

```
chat.converse()
```

A simple conversation :

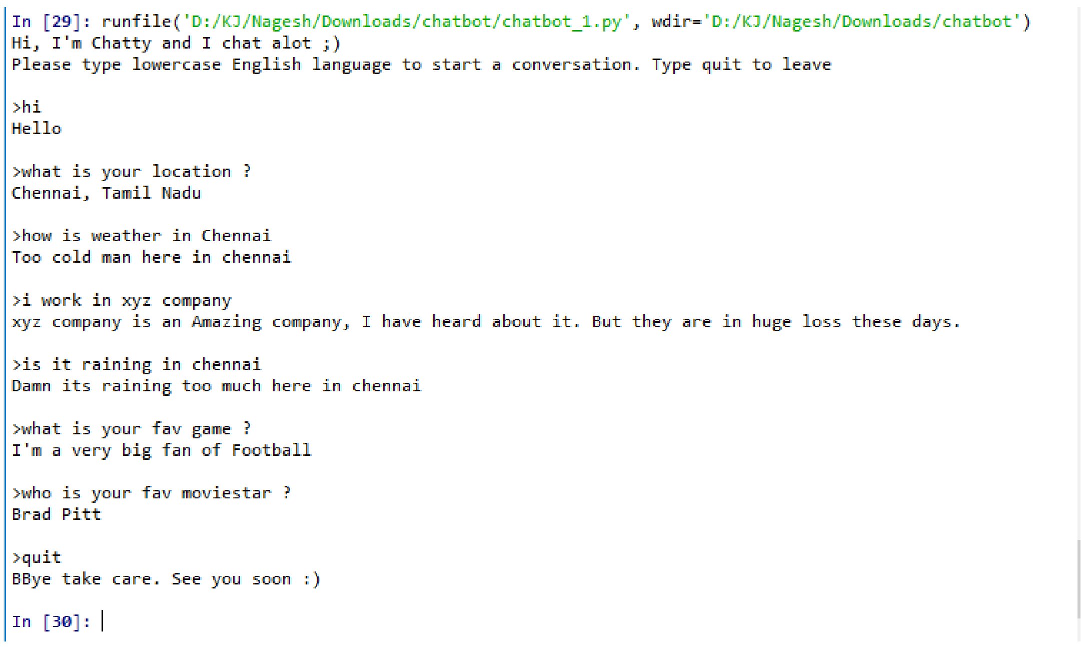


```
In [27]: runfile('D:/KJ/Nagesh/Downloads/chatbot/chatbot_1.py', wdir='D:/KJ/Nagesh/Downloads/chatbot')
Hi, I'm Chatty and I chat alot ;)
Please type lowercase English language to start a conversation. Type quit to leave

>hi
Hey there

>how are you ?
I'm doing good
How about You ?

>i'm also doing good
Nice to hear that

>what is your name ?
My name is Chatty and I'm a chatbot

>what is your age ?
I'm a computer program dude
Seriously you are asking me this?

>sorry Chatty
Its OK, never mind

>
```

a simple conversation with Chatty

As you can see we have just hardcoded the probable question and answers in the list pairs.

Lets interact more with Chatty :



```
In [29]: runfile('D:/KJ/Nagesh/Downloads/chatbot/chatbot_1.py', wdir='D:/KJ/Nagesh/Downloads/chatbot')
Hi, I'm Chatty and I chat alot ;)
Please type lowercase English language to start a conversation. Type quit to leave

>hi
Hello

>what is your location ?
Chennai, Tamil Nadu

>how is weather in Chennai
Too cold man here in chennai

>i work in xyz company
xyz company is an Amazing company, I have heard about it. But they are in huge loss these days.

>is it raining in chennai
Damn its raining too much here in chennai

>what is your fav game ?
I'm a very big fan of Football

>who is your fav moviestar ?
Brad Pitt

>quit
BBye take care. See you soon :)

In [30]: 
```

a simple conversation with Chatty

The nltk.chat chatbots work on the regex of keywords present in your question. So you can add any number of questions in a proper format so that your chatbot doesn't get confused in determining the regex.

In this blog I have explained in simple steps as to how you can build your own chatbot using NLTK and of course its not an intelligent one.

I hope you guys have enjoyed reading.

Happy Learning !!!

For any doubts/suggestions connect with me over LinkedIn.

Bots Chatbots Chatbot Development Nltk Naturallanguageprocessing

272 claps

7

Nagesh Singh Chauhan

Data Science enthusiast | Big Data | Python | Machine Learning | https://www.linkedin.com/in/nagesh-singh-chauhan-6936bb13b/

Follow

Towards Data Science

Sharing concepts, ideas, and codes.

Follow