⑇ main ▾   End-to-end-Machine-Learning-Projects / Deploy_a_Chatbot_with_Python / Deploy_a_Chatbot_with_Python.ipynb ⧉   Go to file   t   ⋯

Rasel1435 Deploy a Chatbot with Python   aeb9401 · 2 minutes ago   ⟳ History

Preview | Code | Blame   338 lines (338 loc) · 91.4 KB   Raw ⧉ ⤓ | ✎ ▾

Deploy a Chatbot with Python

In this article, I will build and deploy a very simple Artificial Intelligent Chatbot. I will use the flask method to deploy the chatbot and the chatterbot package in python to build a chatbot. For building and deploying a chatbot successfully you should be familiar with flask and HTML frameworks. Yes obviously, you should be very good with Python and most importantly the Chatterbot library in python to build this AI chatbot.

Now, before we build and deploy a chatbot let's go through some basics of what a chatbot is and how it works. If you don't want to go through the basics then you will find the code to build and deploy a chatbot at the end of this article.

What is a Chatbot? How to Deploy a Chatbot?

A Chatbot is a computer program which is programmed to interact with a human. A chatbot is designed to reply to a human user based on their queries and conversations. If you think chatbots are a new technology, you're wrong. The first chatbot, Eliza, was built in 1966 at MIT's Artificial Intelligence Laboratory by Joseph Weizenbaum to mimic human conversations

Now, before we build and deploy a chatbot, you should know what the chatterbot library is, as I will use this library for building our chatbot.

What is the Chatterbot Library in Python?

Chatterbot is a Python library that generates responses for users. It uses a lot of pre-trained machine learning algorithms to give a variety of responses. It's easy to create chatbots using the chatterbot library in Python. The chatbot should be designed to be language-independent. He must be trained in several languages. The chatbot is made up of data provided by the user.

I will not cover the basics of Chatterbot here. If you want to learn the basics of this package you can learn it from here.

How to Deploy a Chatbot?

In python, we have Flask, which is useful in the tasks of web development of any program. In one of my previous articles, I deployed a Machine Learning model using flask, I will use the same method to deploy a chatbot.

As I am going to deploy chatbot as a web application, so it is not possible to deploy it without the use of HTML and CSS as these two are the primary packages when it comes to the tasks of web development. So, I will recommend you to learn HTML and CSS also if you are planning to work as a developer.

Now, let's code to build and deploy a chatbot. I will start with the HTML and CSS part, as it will work as the base of our chatbot, without the base we cannot start with the project. So now let's start with the code:

Index.html

```
In [ ]:
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">
    <link rel="stylesheet" href="/static/style.css">
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
</head>
<body>
    <h1 class="jumbotron text-center">AI Chatbot with Python</h1>
<div class="container">
    <div class="row">
        <div class="col-sm-6 offset-sm-3">
            <div id="chatbox" class="border border-success">
                <p class="botText"><span>Hi! I'm Chatterbot</span></p>
            </div>
            <div id="userInput">
                <input id="textInput" class="form-control" type="text" name="msg" placeholder="Type Your Message Here">
                <input id="buttonInput" class="btn btn-success form-control" type="submit" value="Send">
            </div>
        </div>
    </div>

<script>
    function getResponse() {
        let userText = $("#textInput").val();
        let userHtml = '<p class="userText"><span>' + userText + '</span></p>';
        $("#textInput").val("");
        $("#chatbox").append(userHtml);
        document.getElementById('userInput').scrollIntoView({block: 'start', behavior: 'smooth'});
        $.get("/get", { msg: userText }).done(function(data) {
            var botHtml = '<p class="botText"><span>' + data + '</span></p>';
```

```
                    $( "#chatbox" ).append(botHtml1);
            document.getElementById('userInput').scrollIntoView({block: 'start', behavior: 'smooth'});
        });
        }
            $("#textInput").keypress(function(e) {
            //if enter key is pressed
                if(e.which == 13) {
                    getResponse();
                }
            });
            $("#buttonInput").click(function() {
                getResponse();
            });
            </script>
            <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
            <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script>
            <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"></script>
        </div>
        </body>
        </html>
```

style.css

In [ ]:
```
#textInput {
    border: none;
    border-bottom: 3px solid aqua;
}

.userText {
    color: white;
    font-family: monospace;
    font-size: 17px;
    text-align: right;
    line-height: 30px;
}
.userText span {
    background-color: #009688;
    padding: 10px;
    border-radius: 2px;
}
.botText {
    color: white;
    font-family: monospace;
    font-size: 17px;
    text-align: left;
    line-height: 30px;
}
.botText span {
    background-color: #EF5350;
    padding: 10px;
    border-radius: 2px;
}
```

app.py

In [ ]:
```
#imports
from flask import Flask, render_template, request
from chatterbot import ChatBot
from chatterbot.trainers import ChatterBotCorpusTrainer

app = Flask(__name__)
#create chatbot
englishBot = ChatBot("Chatterbot", storage_adapter="chatterbot.storage.SQLStorageAdapter")
trainer = ChatterBotCorpusTrainer(englishBot)
trainer.train("chatterbot.corpus.english") #train the chatter bot for english

#define app routes
@app.route("/")
def index():
    return render_template("index.html")

@app.route("/get")
#function for the bot response
def get_bot_response():
    userText = request.args.get('msg')
    return str(englishBot.get_response(userText))

if __name__ == "__main__":
    app.run()
```
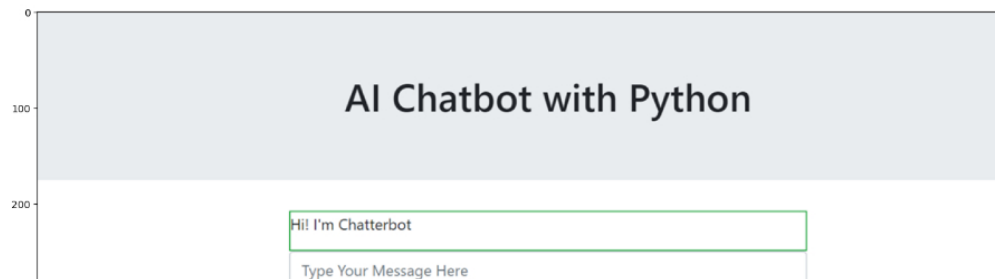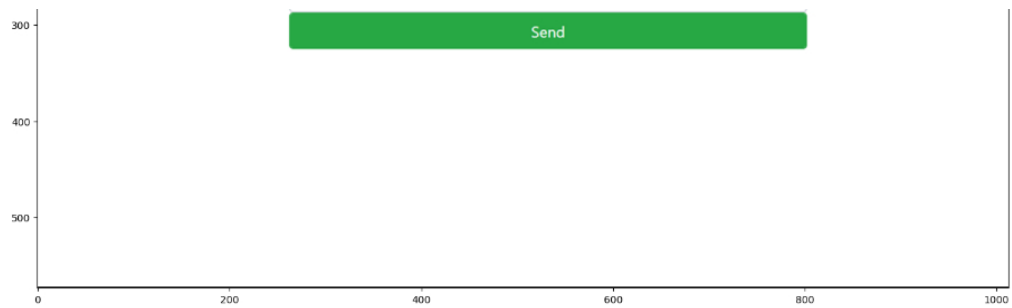
In [12]:
```
import cv2
import matplotlib.pyplot as plt

image = cv2.imread('./images/chatBot.png')
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
plt.figure(figsize=(20,10))
plt.imshow(image)
```

Out[12]: <matplotlib.image.AxesImage at 0x29f57a19e40>

Before running the app.py, make sure that you have included all the files in the same directory. The output of the chatbot is quite good, sometimes you will see some inaccurate results, but most of the times it will work well.

You can work more on the chatbot, the HTML and CSS part will remain the same, so feel free to improve your chatbot. I hope you liked this article on how to build and deploy a chatbot using HTML, CSS and Python. Feel free to ask your valuable questions in the comments section below.

Sheikh Rasel Ahmed

GitHub - https://github.com/Rasel1435

LinkedIn - https://www.linkedin.com/in/shekhnirob1

Facebook - https://www.facebook.com/rasel1435

Instagram - https://www.instagram.com/shekh_nirob

YouTube - https://www.youtube.com/@codewithsheikhrasel