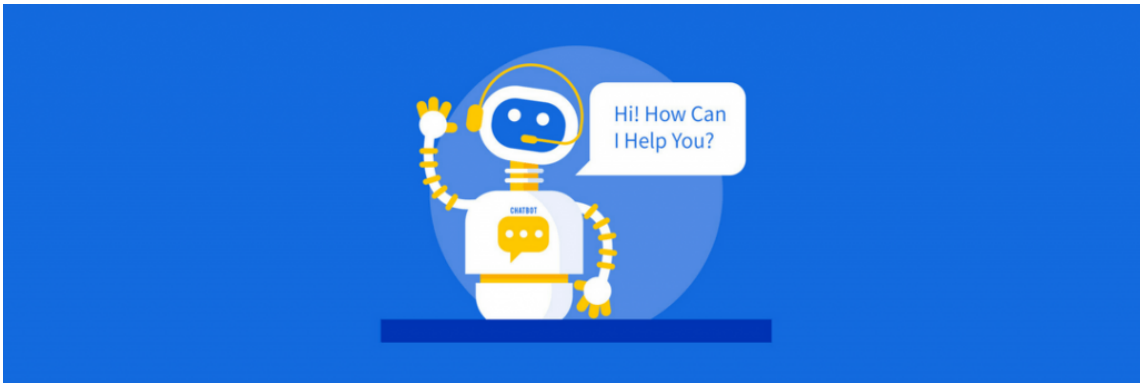
Lab 1. Build a Chatbot using Python/Flask



This lab shows how to create a simple chatbot in Python & Flask using the ChatterBot library. Our bot will be used for small talk, as well as to answer some math questions. Here, we'll scratch the surface of what's possible in building custom chatbots and NLP in general.

Let's talk about Chatterbot, with the help of which we are planning to build our chatbot using Python/Flask.

Lab Solution

Complete solution of this lab is present in following directory:

`/root/Desktop/chatbots-development/Lab01`

Let's build our chatbot

Installing dependencies

```
pip3 install chatterbot chatterbot_corpus
```

Importing Classes – Getting started!!

```
from chatterbot import ChatBot
from chatterbot.trainers import ChatterBotCorpusTrainer
from chatterbot.trainers import ListTrainer
```

Creating the bot

We are creating a Flask app, to get started with Flask, you can visit ***here***

```
app = Flask(__name__)
#bot = ChatBot("Pikachu")
```

We can create and train the bot by creating an instance of ListTrainer and supplying it with the lists of strings:

```
trainer = ListTrainer(bot)
```

Getting started with the training part, there are different ways how we can train the bot, by this,

```
trainer.train(['What is your name?', 'My name is Pikachu'])
trainer.train(['How are you?', 'I am good' ])
trainer.train(['Bye?', 'Bye, see you later' ])
```

or, we can also train by this,

```
conversation = [
    "Hello",
    "Hello!!",
    "How are you doing?",
    "I'm doing great.",
    "That is good to hear",
    "Thank you.",
    "You're welcome."
]

trainer.train(conversation)
```

Training the Bot with corpus of data

You can use your own or an existing corpus of data to train a bot. For example, you can use some corpus provided by chatterbot (inbuilt features):

```
corpus_trainer = ChatterBotCorpusTrainer(bot)
corpus_trainer.train('chatterbot.corpus.english')
```

The **run()** method of Flask class runs the application on the local development server.

```
@app.route("/")
def home():
    return render_template("home.html")
@app.route("/get")
def get_bot_response():
    userText = request.args.get('msg')
    return str(bot.get_response(userText))
if __name__ == "__main__":
    app.run()
```

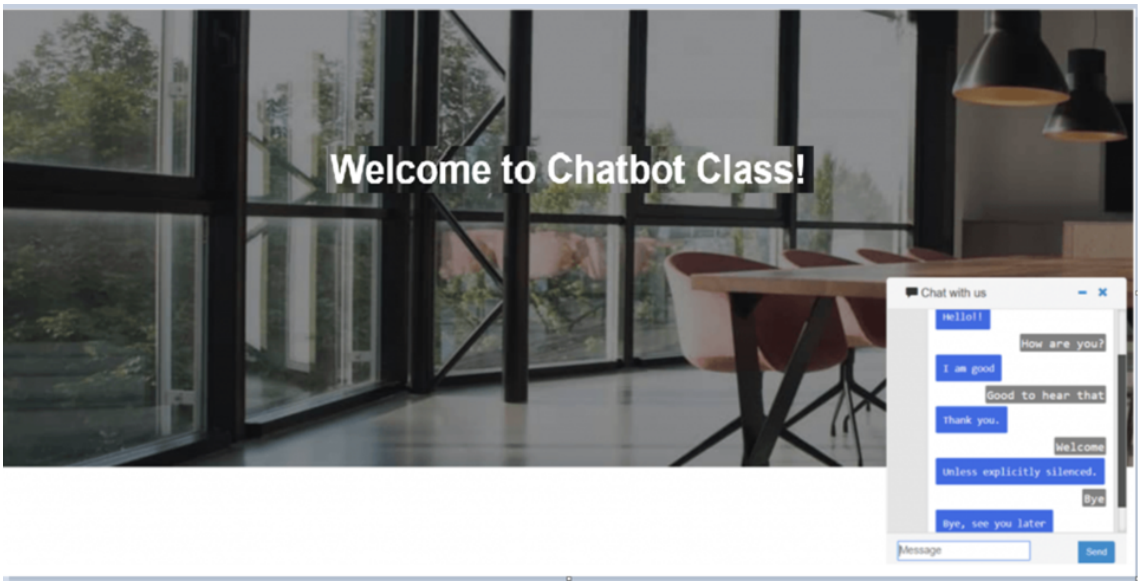
Yay, our first model is ready, let's test our bot.
The above given **Python** script is executed from Python shell. \

Go to terminal, and run the below command:

```
python app.py
```

Below message in Python shell is seen, which indicates that our App is now hosted at http://127.0.0.1:5000/ or localhost:5000

```
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```



References:

- [Chatterbot]