

| NAME          | REGISTRATION NO | ROLL NO | SECTION | GROUP NO. |
|---------------|-----------------|---------|---------|-----------|
| Sahil         | 11802579        | 58      | K18AW   | 20        |
| Mohd Arbaz    | 11802389        | 59      | K18AW   | 20        |
| K. DwarakNath | 11802392        | 60      | K18AW   | 20        |
|               |                 |         |         |           |

## Introduction:

A chatbot is an artificial intelligence (AI) program that simulates interactive human conversation by using key pre-calculated user phrases and auditory or text-based signals. Chatbots are frequently used for basic customer service and marketing systems that frequent social networking hubs and instant messaging (IM) clients. They are also often included in operating systems as intelligent virtual assistants.

A chatbot is also known as an artificial conversational entity (ACE), chat robot, talk bot, chatterbot or chatterbox.

A chatbot is a piece of technology that allows a computer program to communicate with people just like conversing through text messaging using a natural language, say English, to accomplish specific tasks. A chatbot is also known as an artificial conversational entity (ACE), chat robot, talk bot, chatterbot or chatterbox.

Natural Language Processing – Chatbots leverage natural language processing (NLP), which allows the bot to utilize a contextual understanding of a question towards its resolution. It allows you to understand and extract meaningful information (called entities) out of the messages people send. One can use these entities to identify intent, automate some replies, route the conversation to a human via live chat, and collect audience data.

## How do the Chatbots function?

The main technology that lies behind chatbots is NLP and Machine Learning.

When a question is presented to a chatbot, a series of complex algorithms process the received input, understand what the user is asking, and based on that, determines the answer suitable to the question.

Chatbots have to rely on the ability of the algorithms to detect the complexity of both text and spoken words. Some chatbots perform very well to the point it becomes difficult to differentiate whether the user is a machine or a human.

However, handling complex conversations is a huge challenge; where there is a usage of various figures of speech, it may be difficult for machines to understand.

## Use Cases of Chatbots

There are various interesting chatbots, which can make your life easy. For companies, chatbot development focuses around improving their business processes and provide better user experience to their customers.

It is also being utilized to serve customers on social media platforms like Facebook and others. However, most of the Facebook bots are easy to develop and use, as many of them do not need coding, and anyone can create it.

One of the popular modern chatbots is Joy specifically developed by Danny Freed to track and improve mental health. Joy is developed to be a friend via Facebook Messenger.

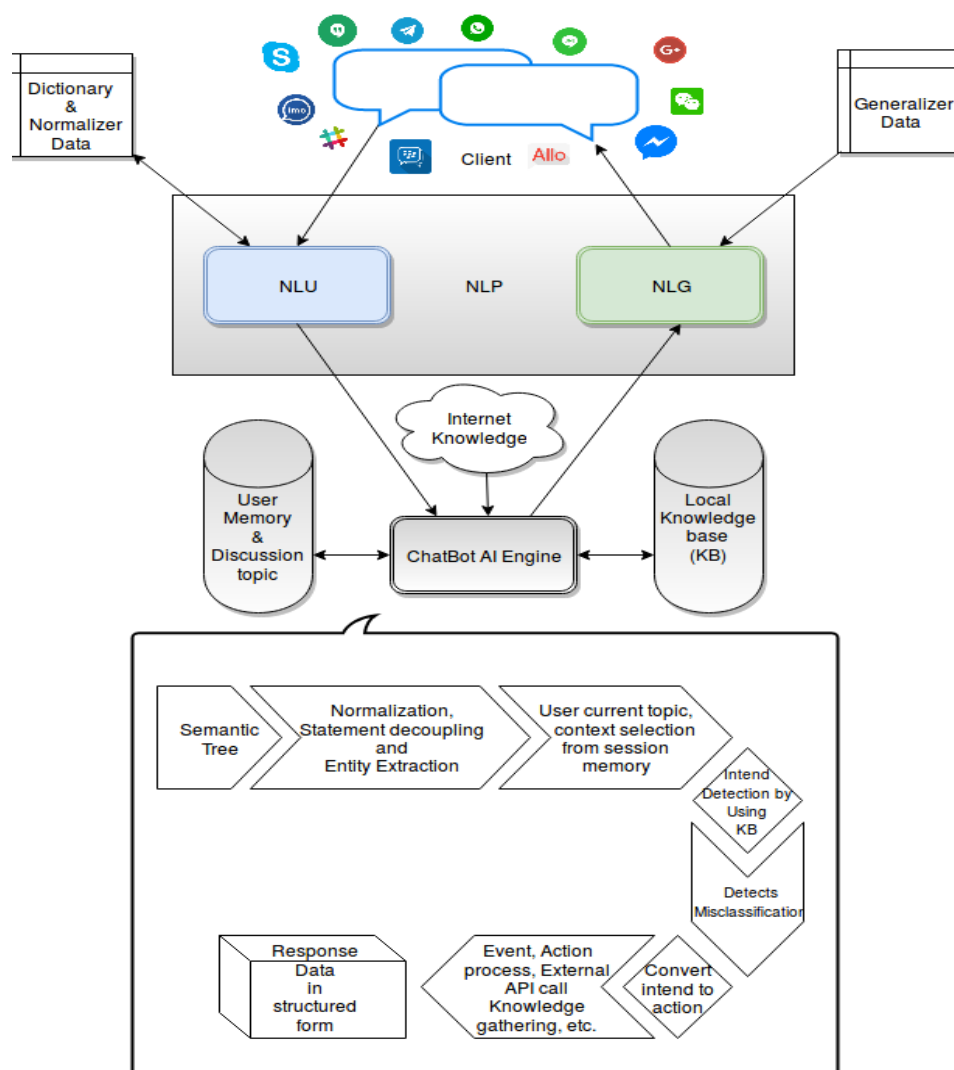
It sends daily check-ins, and offers tips on dealing with different emotional experiences, like anxiety, helping you make life more enjoyable. *The inspiration behind developing Joy came from a close friend of Freed who committed suicide.*

For now, Joy can only ask questions and can generate weekly reports of your mood based on the interactions you have with it. Soon, you will be able to derive long-term solutions and therapies aimed at improving your mental health based on interactions over time.

Let's take a look at some of the chatbots that are used in various industry segments.

Online Shopping, Health Care support etc.

### Data Flow Diagram:



## Libraries:

- **numpy**
- **tflearn**
- **tensorflow**
- **random**
- **json**
- **pickle**
- **nltk**

## code:

```
import nltk
from nltk.stem.lancaster import LancasterStemmer

stemmer = LancasterStemmer()

import numpy
import tflearn
import tensorflow
import random
import json
import pickle

with open("intents.json") as file:
    data = json.load(file)

try:
    with open("data.pickle", "rb") as f:
        words, labels, training, output = pickle.load(f)
except:
    words = []
    labels = []
    docs_x = []
    docs_y = []

    for intent in data["intents"]:
        for pattern in intent["patterns"]:
            wrds = nltk.word_tokenize(pattern)
            words.extend(wrds)
            docs_x.append(wrds)
            docs_y.append(intent["tag"])
```

```

        if intent["tag"] not in labels:
            labels.append(intent["tag"])

words = [stemmer.stem(w.lower()) for w in words if w != "?"]
words = sorted(list(set(words)))

labels = sorted(labels)

training = []
output = []

out_empty = [0 for _ in range(len(labels))]

for x, doc in enumerate(docs_x):
    bag = []

    wrds = [stemmer.stem(w.lower()) for w in doc]

    for w in words:
        if w in wrds:
            bag.append(1)
        else:
            bag.append(0)

    output_row = out_empty[:]
    output_row[labels.index(docs_y[x])] = 1

    training.append(bag)
    output.append(output_row)

training = numpy.array(training)
output = numpy.array(output)

with open("data.pickle", "wb") as f:
    pickle.dump((words, labels, training, output), f)

tensorflow.reset_default_graph()

net = tflearn.input_data(shape=[None, len(training[0])])
net = tflearn.fully_connected(net, 8)
net = tflearn.fully_connected(net, 8)
net = tflearn.fully_connected(net, len(output[0]), activation="softmax")
net = tflearn.regression(net)

model = tflearn.DNN(net)

try:

```

```

    model.load("model.tflearn")
except:
    tensorflow.reset_default_graph()

    net = tflearn.input_data(shape=[None, len(training[0])])
    net = tflearn.fully_connected( net, 8)
    net = tflearn.fully_connected( net, 8)
    net = tflearn.fully_connected( net, len(output[0]), activation="softmax")
    net = tflearn.regression(net)

    model = tflearn.DNN(net)
    model.fit(training, output, n_epoch=1000, batch_size=8, show_metric=True)
    model.save("model.tflearn")

def bag_of_words(s, words):
    bag = [0 for _ in range(len(words))]

    s_words = nltk.word_tokenize(s)
    s_words = [stemmer.stem(word.lower()) for word in s_words]

    for se in s_words:
        for i, w in enumerate(words):
            if w == se:
                bag[i] = 1

    return numpy.array(bag)

def chat():
    print("Start talking with the bot (type quit to stop)!")
    while True:
        inp = input("You: ")
        if inp.lower() == "quit":
            break

        results = model.predict([bag_of_words(inp, words)])
        results_index = numpy.argmax(results)
        tag = labels[results_index]

        for tg in data["intents"]:
            if tg['tag'] == tag:
                responses = tg['responses']

        print("Bot:", random.choice(responses))

chat()

```

# Code Snippet:

```
Command Prompt - main.py
C:\Users\Abhikush>cd desktop

C:\Users\Abhikush\Desktop>cd college enquiry bot

C:\Users\Abhikush\Desktop\College Enquiry bot>main.py
C:\Users\Abhikush\AppData\Local\Programs\Python\Python36\lib\site-packages\tensorflow\python\framework\dtypes.py:469: FutureWarning: Passing (type, 1) or 'ltype' as a synonym of type is deprecated; in a future v
ersion of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  np.qint8 = np.dtype(("qint8", np.int8, 1))
C:\Users\Abhikush\AppData\Local\Programs\Python\Python36\lib\site-packages\tensorflow\python\framework\dtypes.py:470: FutureWarning: Passing (type, 1) or 'ltype' as a synonym of type is deprecated; in a future v
ersion of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  np.quint8 = np.dtype(("quint8", np.uint8, 1))
C:\Users\Abhikush\AppData\Local\Programs\Python\Python36\lib\site-packages\tensorflow\python\framework\dtypes.py:471: FutureWarning: Passing (type, 1) or 'ltype' as a synonym of type is deprecated; in a future v
ersion of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  np.qint16 = np.dtype(("qint16", np.int16, 1))
C:\Users\Abhikush\AppData\Local\Programs\Python\Python36\lib\site-packages\tensorflow\python\framework\dtypes.py:472: FutureWarning: Passing (type, 1) or 'ltype' as a synonym of type is deprecated; in a future v
ersion of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  np.quint16 = np.dtype(("quint16", np.uint16, 1))
C:\Users\Abhikush\AppData\Local\Programs\Python\Python36\lib\site-packages\tensorflow\python\framework\dtypes.py:473: FutureWarning: Passing (type, 1) or 'ltype' as a synonym of type is deprecated; in a future v
ersion of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  np.qint32 = np.dtype(("qint32", np.int32, 1))
C:\Users\Abhikush\AppData\Local\Programs\Python\Python36\lib\site-packages\tensorflow\python\framework\dtypes.py:476: FutureWarning: Passing (type, 1) or 'ltype' as a synonym of type is deprecated; in a future v
ersion of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  np.resource = np.dtype(("resource", np.ubyte, 1))
hdf5 is not supported on this machine (please install/reinstall h5py for optimal experience)
curses is not supported on this machine (please install/reinstall curses for an optimal experience)
Scipy not supported!
2020-04-07 21:07:11.973716: I C:\tf_jenkins\home\workspace\rel-win\Windows\Python36\tensorflow\core\platform\cpu_feature_guard.cc:137] Your CPU supports instructions that this TensorFlow binary was not compiled t
o use: AVX AVX2
Start talking with the bot (type quit to stop)!
You: hello
Bot: Hello!
You: how many course
Bot: There are more than 200 courses in our University
You: faculty
Bot: 2000+ faculty in our University
You: student
Bot: We have more than 30,000 students in our university
You: goodbye
Bot: Goodbye!
You: cya
Bot: Sad to see you go :(
You: bye
Bot: More than 4000 CCTV cameras are present in our university
You: _
```

```
Command Prompt - main.py
C:\Users\Abhikush>cd desktop

C:\Users\Abhikush\Desktop>cd college enquiry bot

C:\Users\Abhikush\Desktop\College Enquiry bot>main.py
C:\Users\Abhikush\AppData\Local\Programs\Python\Python36\lib\site-packages\tensorflow\python\framework\dtypes.py:469: FutureWarning: Passing (type, 1) or 'ltype' as a synonym of type is deprecated; in a future v
ersion of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  np.qint8 = np.dtype(("qint8", np.int8, 1))
C:\Users\Abhikush\AppData\Local\Programs\Python\Python36\lib\site-packages\tensorflow\python\framework\dtypes.py:470: FutureWarning: Passing (type, 1) or 'ltype' as a synonym of type is deprecated; in a future v
ersion of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  np.quint8 = np.dtype(("quint8", np.uint8, 1))
C:\Users\Abhikush\AppData\Local\Programs\Python\Python36\lib\site-packages\tensorflow\python\framework\dtypes.py:471: FutureWarning: Passing (type, 1) or 'ltype' as a synonym of type is deprecated; in a future v
ersion of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  np.qint16 = np.dtype(("qint16", np.int16, 1))
C:\Users\Abhikush\AppData\Local\Programs\Python\Python36\lib\site-packages\tensorflow\python\framework\dtypes.py:472: FutureWarning: Passing (type, 1) or 'ltype' as a synonym of type is deprecated; in a future v
ersion of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  np.quint16 = np.dtype(("quint16", np.uint16, 1))
C:\Users\Abhikush\AppData\Local\Programs\Python\Python36\lib\site-packages\tensorflow\python\framework\dtypes.py:473: FutureWarning: Passing (type, 1) or 'ltype' as a synonym of type is deprecated; in a future v
ersion of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  np.qint32 = np.dtype(("qint32", np.int32, 1))
C:\Users\Abhikush\AppData\Local\Programs\Python\Python36\lib\site-packages\tensorflow\python\framework\dtypes.py:476: FutureWarning: Passing (type, 1) or 'ltype' as a synonym of type is deprecated; in a future v
ersion of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  np.resource = np.dtype(("resource", np.ubyte, 1))
hdf5 is not supported on this machine (please install/reinstall h5py for optimal experience)
curses is not supported on this machine (please install/reinstall curses for an optimal experience)
Scipy not supported!
2020-04-07 21:07:11.973716: I C:\tf_jenkins\home\workspace\rel-win\Windows\Python36\tensorflow\core\platform\cpu_feature_guard.cc:137] Your CPU supports instructions that this TensorFlow binary was not compiled t
o use: AVX AVX2
Start talking with the bot (type quit to stop)!
You: _
```

```
intents.json - Notepad
File Edit Format View Help
{"intents": [
  {
    "tag": "greeting",
    "patterns": ["Hi", "How are you", "Is anyone there?", "Hello", "Good day", "Whats up"],
    "responses": ["Hello!", "Good to see you again!", "Hi there, how can I help?"],
    "context_set": ""
  },
  {
    "tag": "goodbye",
    "patterns": ["cya", "See you later", "Goodbye", "I am Leaving", "Have a Good day"],
    "responses": ["Sad to see you go :( ", "Talk to you later", "Goodbye!"],
    "context_set": ""
  },
  {
    "tag": "courses",
    "patterns": ["how many", "how many courses are available", "how many courses your university provide", "courses?"],
    "responses": ["There are more than 200 courses in our University"],
    "context_set": ""
  },
  {
    "tag": "faculty",
    "patterns": ["number of faculty?", "How many faculty are", "How many faculty in your university?", "faculty?"],
    "responses": ["More than 2000 faculty work in our University", "2000+ faculty in our University"],
    "context_set": ""
  },
  {
    "tag": "camera",
    "patterns": ["How many camera?", "camera installed in your University", "camera?", "cctv?", "cctv", "How many camera in your university"],
    "responses": ["More than 4000 CCTV cameras are present in our university"],
    "context_set": ""
  },
  {
    "tag": "students",
    "patterns": ["How many students in your university", "strength of students", "students?"],
    "responses": ["We have more than 30,000 students in our university"],
    "context_set": ""
  }
]
}
```

## Conclusion:

Smart solutions are important for the success of any business. From providing 24/7 customer service, improving current marketing activities, saving time spent on engaging with users to improving internal processes, chatbots can yield the much-needed competitive advantage. If you are looking to develop a chatbot, the best thing to do is to approach a company that will understand your business needs to develop a chatbot that helps you achieve your business goals.