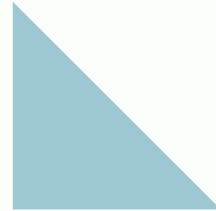
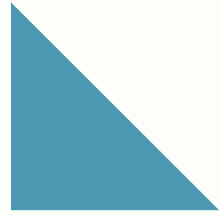


July 2021

# FAQ CHATBOT

Prepared by: Srivathsan



# About

An Covid19 FAQ (Frequently Asked Questions) ChatBot is a type of internet bot or software application that is beneficial for answering some of the most frequently asked questions related to covid19. This FAQ bots helps the customer's be aware about the pandemic.

## Mission and Vision

Never in the history of public health, a pandemic of a disease threatened the humanity as COVID-19, technical name of a newly identified coronavirus, has inflicted. The disease is caused by the SARS-coronavirus-2, a virus primarily zoonotic and was not found in humans. WHO has declared COVID-19 a global pandemic and a public health emergency. The spread of corona epidemic is unprecedented and has reached 199 countries and territories around the world (and the cruise ship Diamond Princess harbored in Yokohama, Japan), and has affected over 556,141 people, testing positive for coronavirus. The death toll has reached 25,237 (Worldometer, March 27, 2020; 14.36 GMT). Pandemic has spread with a high velocity across the globe within a short period of time. Unlike SARS (2003) and MERS (2012), the case fatality rate is higher at 2–3%. In order to educate the public about this serious threat, this chatbot is manufactured to create an awareness amongst the public.

# Concept

Since we are going to develop a deep learning based model, we need data to train our model. So we can just create our own dataset in order to train the model. So an intent is created in a JSON file format. The concept here is to define different intents and make the training sample for those intents and train the chatbot model with those training sample data as model training data(X) and intents as model training category(Y)

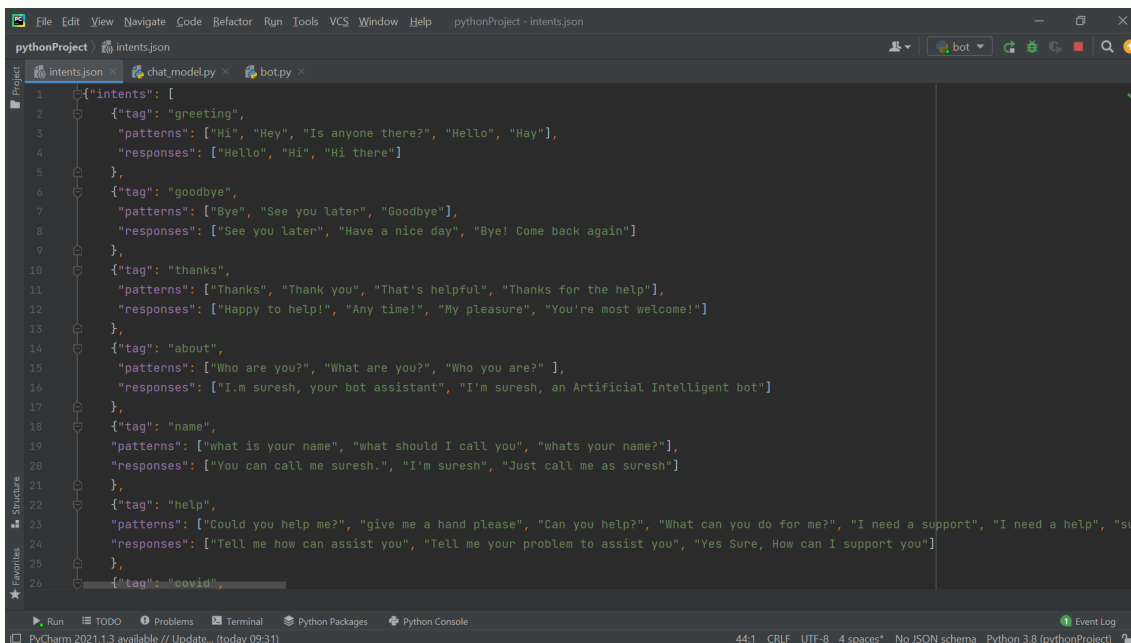
# Implementation

The packages that are used are:

- Tensorflow
- NLTK
- Colorama
- Numpy
- Scikit-Learn

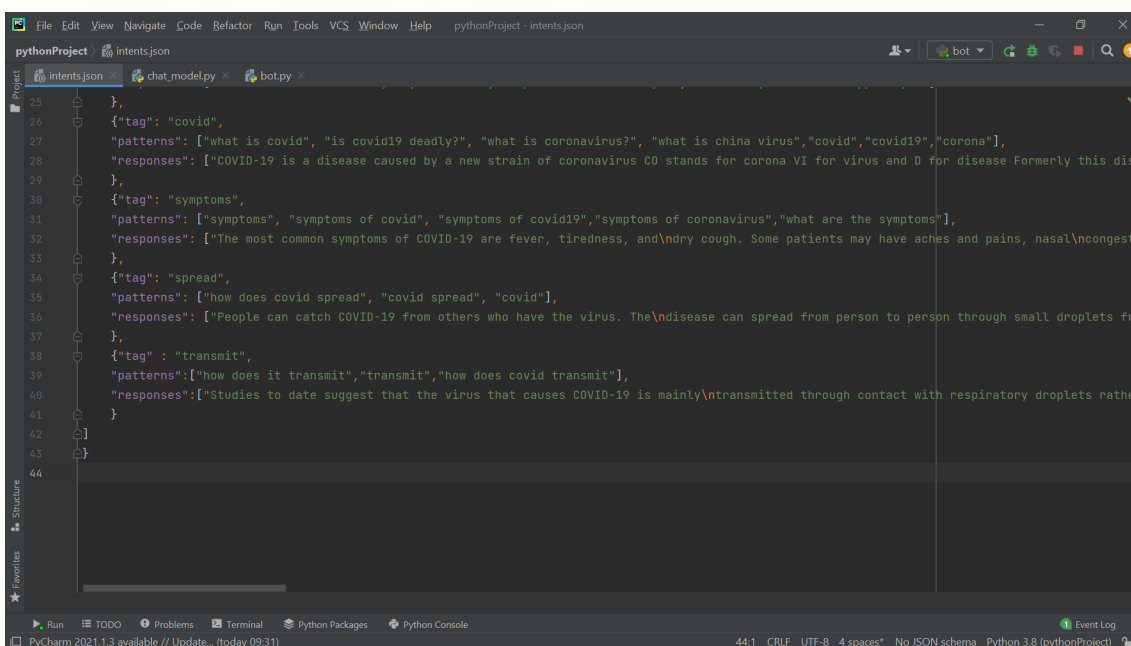
# Intents

Few simple intents were defined and bunch of messages that corresponds to those intents and also mapped some responses according to each intent category. Created a JSON file named "intents.json" including these data



```

1  {"intents": [
2    {
3      "tag": "greeting",
4      "patterns": ["Hi", "Hey", "Is anyone there?", "Hello", "Hay"],
5      "responses": ["Hello", "Hi", "Hi there"]
6    },
7    {
8      "tag": "goodbye",
9      "patterns": ["Bye", "See you later", "Goodbye"],
10     "responses": ["See you later", "Have a nice day", "Bye! Come back again"]
11   },
12   {
13     "tag": "thanks",
14     "patterns": ["Thanks", "Thank you", "That's helpful", "Thanks for the help"],
15     "responses": ["Happy to help!", "Any time!", "My pleasure", "You're most welcome!"]
16   },
17   {
18     "tag": "about",
19     "patterns": ["Who are you?", "What are you?", "Who you are?" ],
20     "responses": ["I'm suresh, your bot assistant", "I'm suresh, an Artificial Intelligent bot"]
21   },
22   {
23     "tag": "name",
24     "patterns": ["What is your name", "what should I call you", "whats your name?"],
25     "responses": ["You can call me suresh.", "I'm suresh", "Just call me as suresh"]
26   },
27   {
28     "tag": "help",
29     "patterns": ["Could you help me?", "give me a hand please", "Can you help?", "What can you do for me?", "I need a support", "I need a help", "sup"],
30     "responses": ["Tell me how can assist you", "Tell me your problem to assist you", "Yes Sure, How can I support you"]
31   },
32   {
33     "tag": "covid",
  
```



```

34   {
35     "tag": "covid",
36     "patterns": ["what is covid", "is covid19 deadly?", "what is coronavirus?", "what is china virus", "covid", "covid19", "corona"],
37     "responses": ["COVID-19 is a disease caused by a new strain of coronavirus CO stands for corona VI for virus and D for disease Formerly this disease was called 2019-nCoV. It is a new virus. It is a new virus. It is a new virus."],
38   },
39   {
40     "tag": "symptoms",
41     "patterns": ["symptoms", "symptoms of covid", "symptoms of covid19", "symptoms of coronavirus", "what are the symptoms"],
42     "responses": ["The most common symptoms of COVID-19 are fever, tiredness, and dry cough. Some patients may have aches and pains, nasal congestion, runny nose, sore throat, and loss of taste or smell."],
43   },
44   {
45     "tag": "spread",
46     "patterns": ["how does covid spread", "covid spread", "covid"],
47     "responses": ["People can catch COVID-19 from others who have the virus. The disease can spread from person to person through small droplets from the nose or mouth when coughing or sneezing, or from surfaces and objects."],
48   },
49   {
50     "tag": "transmit",
51     "patterns": ["how does it transmit", "transmit", "how does covid transmit"],
52     "responses": ["Studies to date suggest that the virus that causes COVID-19 is mainly transmitted through contact with respiratory droplets rather than through the air."],
53   },
54 ]
55 }
  
```

# Preparing the data

After importing all the required packages, We have to load the JSON file and extract the data

```
File Edit View Navigate Code Refactor Run Tools VCS Window Help pythonProject - chat_model.py
pythonProject > chat_model.py
intents.json x chat_model.py x bot.py x
import json
import numpy as np
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Embedding, GlobalAveragePooling1D
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
from sklearn.preprocessing import LabelEncoder
with open('intents.json') as file:
    data = json.load(file)
    training_sentences = []
    training_labels = []
    labels = []
    responses = []
for intent in data['intents']:
    for pattern in intent['patterns']:
        training_sentences.append(pattern)
        training_labels.append(intent['tag'])
    responses.append(intent['responses'])
    if intent['tag'] not in labels:
        labels.append(intent['tag'])
num_classes = len(labels)
```

The variable "*training\_sentences*" holds all the training data (which are the sample messages in each intent) and the "*training\_labels*" variable holds all the target labels correspond to each training data. Then we use "LabelEncoder()" function is used which is provided by scikit-learn to convert the target labels into a model understandable form.

```
lbl_encoder = LabelEncoder()  
lbl_encoder.fit(training_labels)  
training_labels = lbl_encoder.transform(training_labels)
```

Now, to the part of vectorizing our text data corpus by using the "*Tokenizer*" class. This will allows to limit the vocabulary size up to some defined number.

When using this class for the text pre-processing task, by default all punctuations will be removed, turning the texts into space-separated sequences of words, and these sequences are then split into lists of tokens, which will be indexed or vectorized.

Adding the "oov\_token" which is a value for "out of token" to manage the vocabulary words(tokens) at inference time.

The variable "*pad\_sequences*" is used to make all the training text sequences into the same size.

```
vocab_size = 1000
embedding_dim = 16
max_len = 20
oov_token = "<OOV>"
tokenizer = Tokenizer(num_words=vocab_size, oov_token=oov_token) # adding out of vocabulary token
tokenizer.fit_on_texts(training_sentences)
word_index = tokenizer.word_index
sequences = tokenizer.texts_to_sequences(training_sentences)
padded_sequences = pad_sequences(sequences, truncating='post', maxlen=max_len)
```

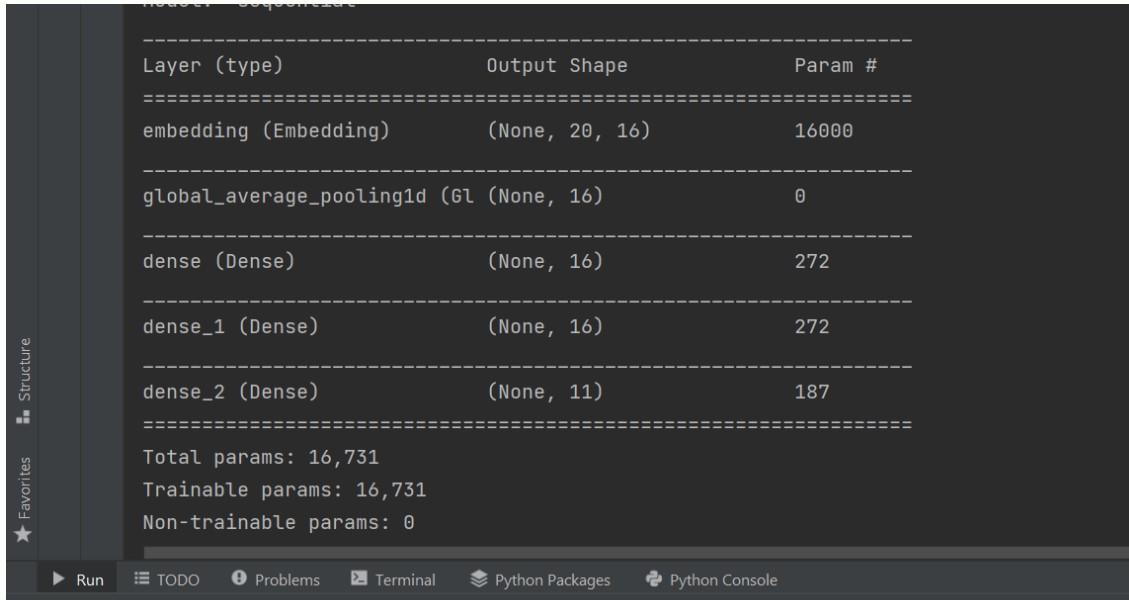
# Training the model

A neural network architecture is defined for the model.

A sequential model class of keras is used

```
model = Sequential()
model.add(Embedding(vocab_size, embedding_dim, input_length=max_len))
model.add(GlobalAveragePooling1D())
model.add(Dense(16, activation='relu'))
model.add(Dense(16, activation='relu'))
model.add(Dense(num_classes, activation='softmax'))
model.compile(loss='sparse_categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
model.summary()
```

This is our model's architecture :



The screenshot shows a Jupyter Notebook interface with a sidebar on the left containing 'Structure' and 'Favorites' tabs. The main area displays the model architecture summary for a 'model' object of type 'Sequential'.

Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 20, 16)	16000
global_average_pooling1d (GlobalAveragePooling1D)	(None, 16)	0
dense (Dense)	(None, 16)	272
dense_1 (Dense)	(None, 16)	272
dense_2 (Dense)	(None, 11)	187

Summary statistics:

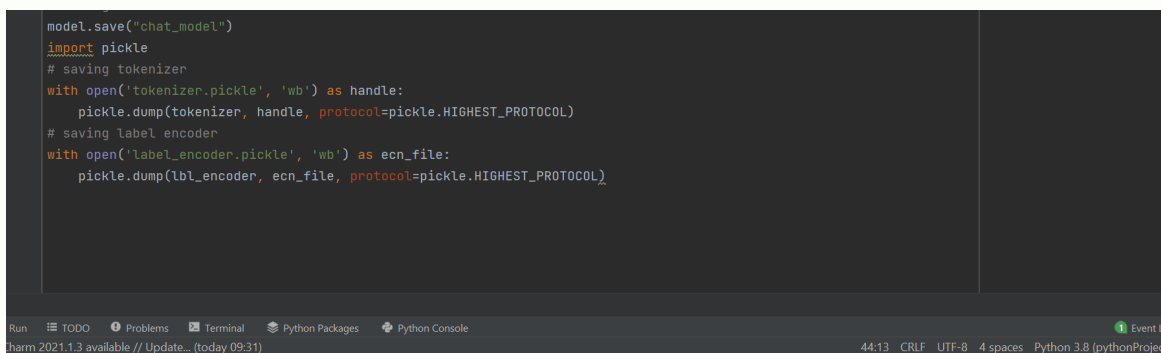
- Total params: 16,731
- Trainable params: 16,731
- Non-trainable params: 0

The bottom of the screenshot shows the Jupyter Notebook toolbar with buttons for 'Run', 'TODO', 'Problems', 'Terminal', 'Python Packages', and 'Python Console'.

Model is fitted using the training data's and labels

```
epochs = 550
history = model.fit(padded_sequences, np.array(training_labels), epochs=epochs)
```

## Saving the model



The screenshot shows a Jupyter Notebook interface with a code cell containing the following Python code:

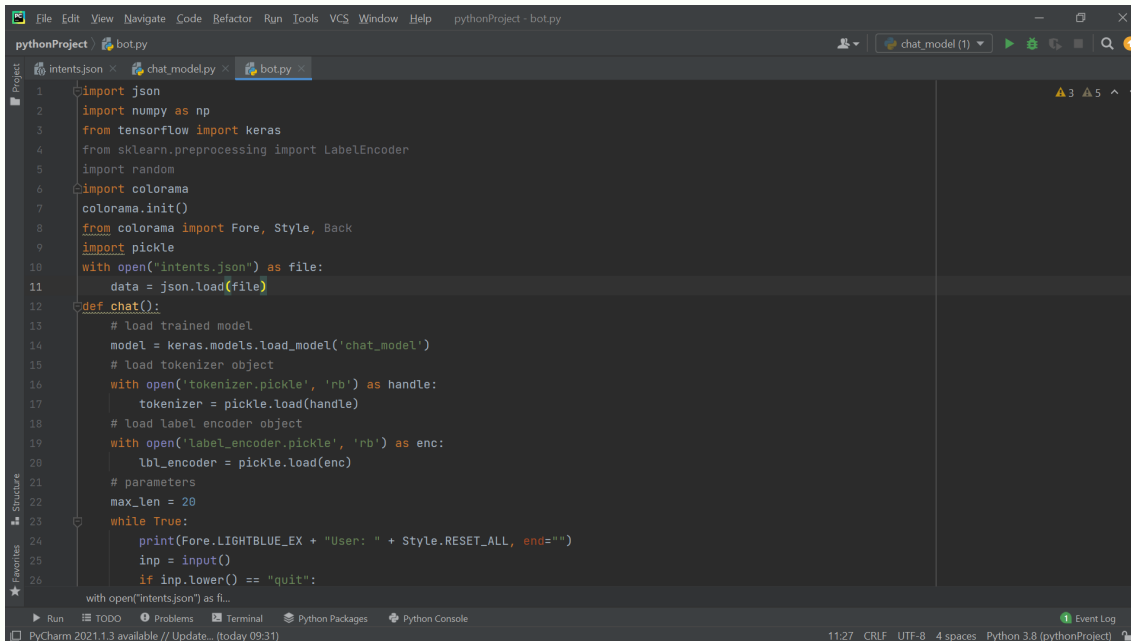
```
model.save("chat_model")
import pickle
# saving tokenizer
with open('tokenizer.pickle', 'wb') as handle:
    pickle.dump(tokenizer, handle, protocol=pickle.HIGHEST_PROTOCOL)
# saving label encoder
with open('label_encoder.pickle', 'wb') as ecn_file:
    pickle.dump(lbl_encoder, ecn_file, protocol=pickle.HIGHEST_PROTOCOL)
```

The bottom of the screenshot shows the Jupyter Notebook toolbar with buttons for 'Run', 'TODO', 'Problems', 'Terminal', 'Python Packages', and 'Python Console'. The status bar at the bottom indicates 'Python 3.8 (pythonProject)' and '44:13 CRLF UTF-8 4 spaces'.

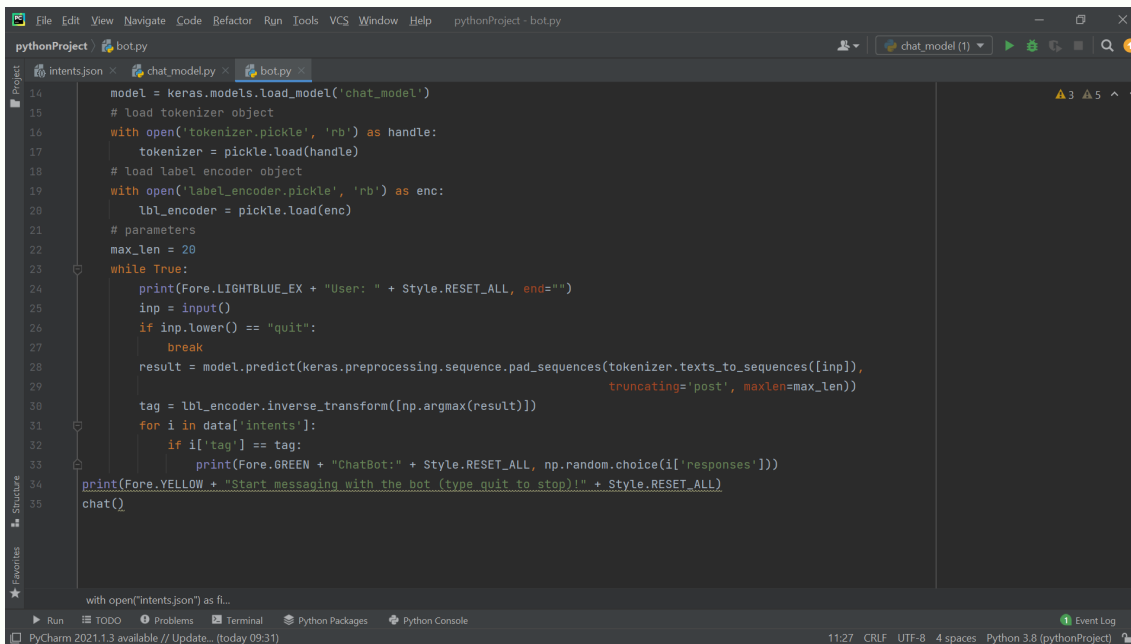


# Implementing a chat function

A chat function is created in which when a message is received from the user the chat bot will calculate the similarity between the test sequence and training data

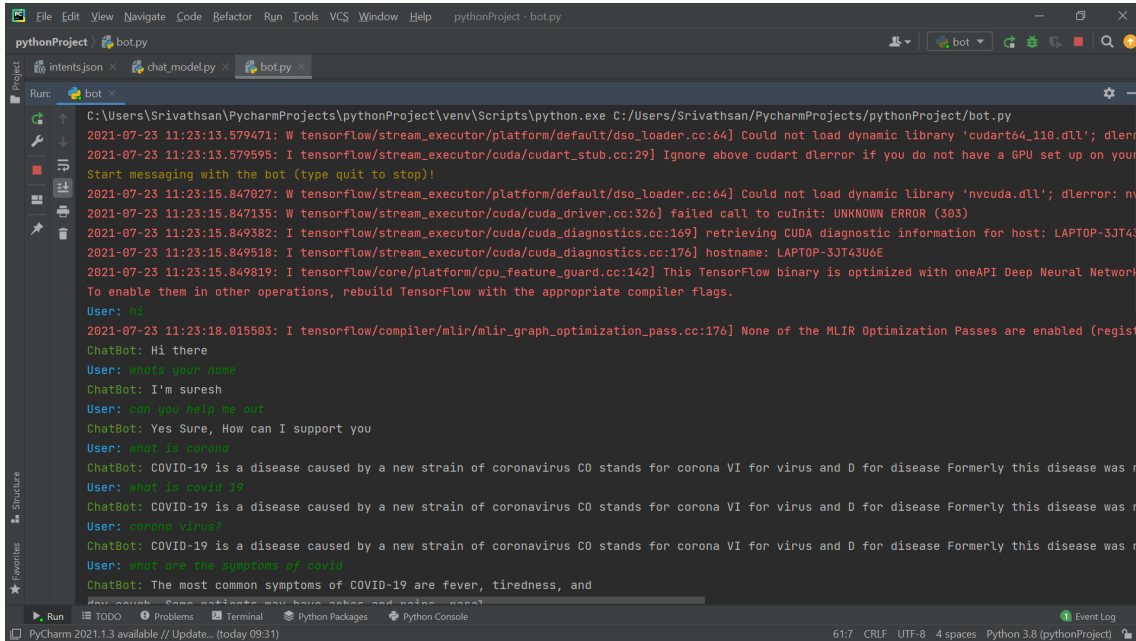


```
pythonProject - bot.py
1 import json
2 import numpy as np
3 from tensorflow import keras
4 from sklearn.preprocessing import LabelEncoder
5 import random
6 import colorama
7 colorama.init()
8 from colorama import Fore, Style, Back
9 import pickle
10 with open("intents.json") as file:
11     data = json.load(file)
12 def chat():
13     # load trained model
14     model = keras.models.load_model('chat_model')
15     # load tokenizer object
16     with open('tokenizer.pickle', 'rb') as handle:
17         tokenizer = pickle.load(handle)
18     # load label encoder object
19     with open('label_encoder.pickle', 'rb') as enc:
20         lbl_encoder = pickle.load(enc)
21     # parameters
22     max_len = 20
23     while True:
24         print(Fore.LIGHTBLUE_EX + "User: " + Style.RESET_ALL, end="")
25         inp = input()
26         if inp.lower() == "quit":
27             with open("intents.json") as fi...
```

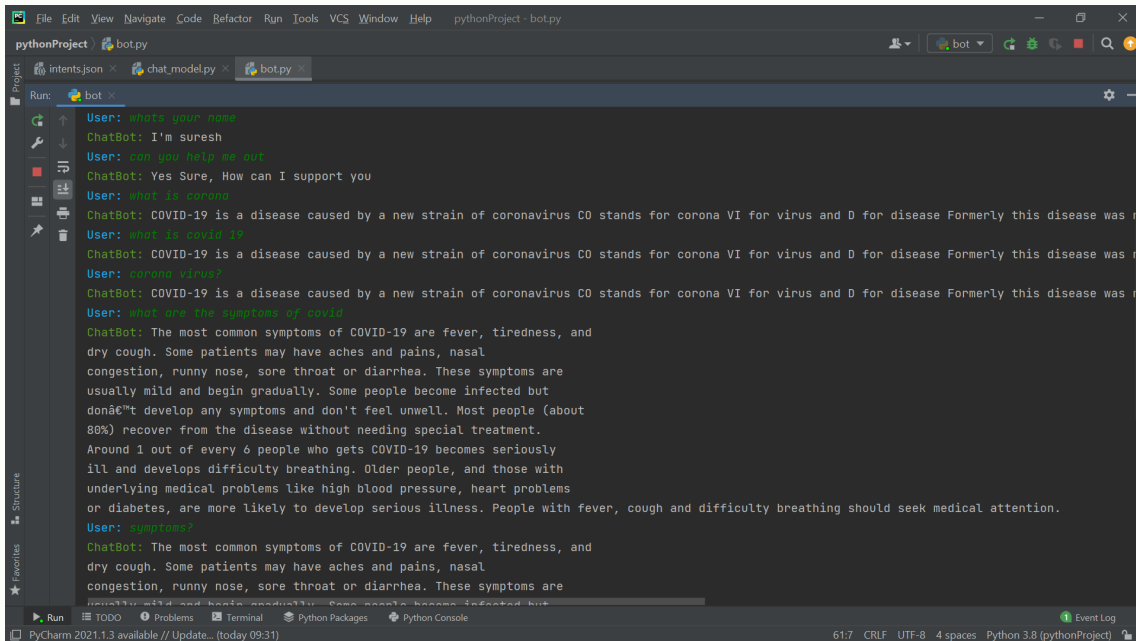


```
pythonProject - bot.py
14 model = keras.models.load_model('chat_model')
15 # load tokenizer object
16 with open('tokenizer.pickle', 'rb') as handle:
17     tokenizer = pickle.load(handle)
18 # load label encoder object
19 with open('label_encoder.pickle', 'rb') as enc:
20     lbl_encoder = pickle.load(enc)
21 # parameters
22 max_len = 20
23 while True:
24     print(Fore.LIGHTBLUE_EX + "User: " + Style.RESET_ALL, end="")
25     inp = input()
26     if inp.lower() == "quit":
27         break
28     result = model.predict(keras.preprocessing.sequence.pad_sequences(tokenizer.texts_to_sequences([inp]),
29                                                                     truncating='post', maxlen=max_len))
30     tag = lbl_encoder.inverse_transform([np.argmax(result)])
31     for i in data['intents']:
32         if i['tag'] == tag:
33             print(Fore.GREEN + "ChatBot:" + Style.RESET_ALL, np.random.choice(i['responses']))
34 print(Fore.YELLOW + "Start messaging with the bot (type quit to stop)!" + Style.RESET_ALL)
35 chat()
```

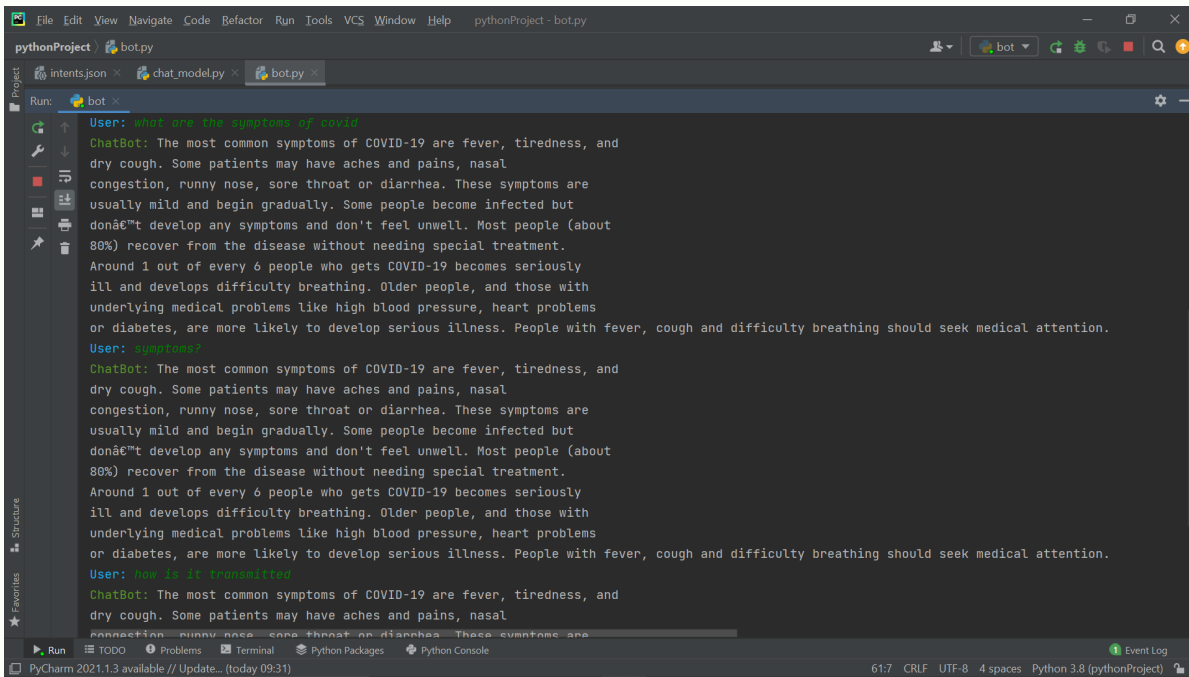
# Output



```
File Edit View Navigate Code Refactor Run Tools VCS Window Help pythonProject - bot.py
pythonProject bot.py
Intents.json chat_model.py bot.py
Run: bot
C:\Users\Srivathsan\PycharmProjects\pythonProject\venv\Scripts\python.exe C:\Users\Srivathsan\PycharmProjects\pythonProject\bot.py
2021-07-23 11:23:13.579471: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'cudart64_110.dll'; dlerror:
2021-07-23 11:23:13.579595: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart dlerror if you do not have a GPU set up on your
Start messaging with the bot (type quit to stop)!
2021-07-23 11:23:15.847027: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'nvcuda.dll'; dlerror: nvcuda
2021-07-23 11:23:15.847135: W tensorflow/stream_executor/cuda/cuda_driver.cc:326] failed call to cuInit: UNKNOWN ERROR (303)
2021-07-23 11:23:15.849382: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:169] retrieving CUDA diagnostic information for host: LAPTOP-3JT43U6E
2021-07-23 11:23:15.849518: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:176] hostname: LAPTOP-3JT43U6E
2021-07-23 11:23:15.849819: I tensorflow/core/platform/cpu_feature_guard.cc:142] This TensorFlow binary is optimized with oneAPI Deep Neural Network
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
User: hi
2021-07-23 11:23:18.015503: I tensorflow/compiler/mlir/mlir_graph_optimization_pass.cc:176] None of the MLIR Optimization Passes are enabled (regist
ChatBot: Hi there
User: whats your name
ChatBot: I'm suresh
User: can you help me out
ChatBot: Yes Sure, How can I support you
User: what is covid
ChatBot: COVID-19 is a disease caused by a new strain of coronavirus C0 stands for corona VI for virus and D for disease Formerly this disease was na
User: what is covid 19
ChatBot: COVID-19 is a disease caused by a new strain of coronavirus C0 stands for corona VI for virus and D for disease Formerly this disease was na
User: whats virus?
ChatBot: COVID-19 is a disease caused by a new strain of coronavirus C0 stands for corona VI for virus and D for disease Formerly this disease was na
User: what are the symptoms of covid
ChatBot: The most common symptoms of COVID-19 are fever, tiredness, and
=====
Run TODO Problems Terminal Python Packages Python Console Event Log
PyCharm 2021.1.3 available // Update... (today 09:31) 61.7 CRLF UTF-8 4 spaces Python 3.8 (pythonProject)
```

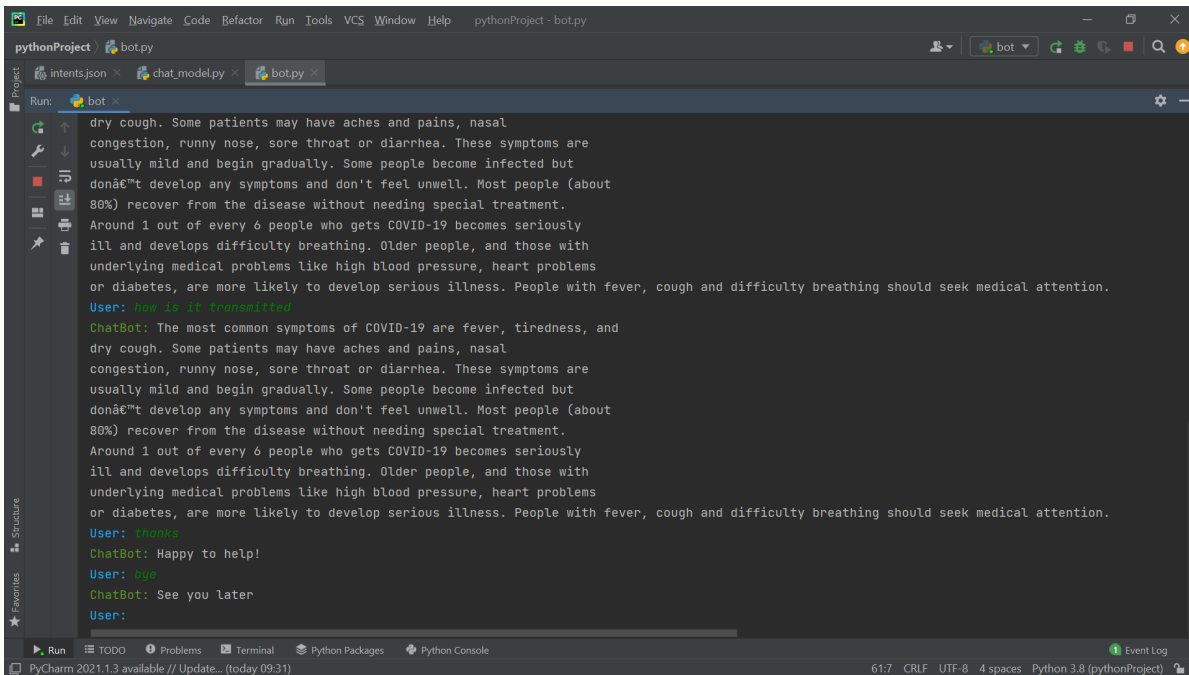


```
File Edit View Navigate Code Refactor Run Tools VCS Window Help pythonProject - bot.py
pythonProject bot.py
Intents.json chat_model.py bot.py
Run: bot
User: whats your name
ChatBot: I'm suresh
User: can you help me out
ChatBot: Yes Sure, How can I support you
User: what is covid
ChatBot: COVID-19 is a disease caused by a new strain of coronavirus C0 stands for corona VI for virus and D for disease Formerly this disease was na
User: what is covid 19
ChatBot: COVID-19 is a disease caused by a new strain of coronavirus C0 stands for corona VI for virus and D for disease Formerly this disease was na
User: whats virus?
ChatBot: COVID-19 is a disease caused by a new strain of coronavirus C0 stands for corona VI for virus and D for disease Formerly this disease was na
User: what are the symptoms of covid
ChatBot: The most common symptoms of COVID-19 are fever, tiredness, and
dry cough. Some patients may have aches and pains, nasal
congestion, runny nose, sore throat or diarrhea. These symptoms are
usually mild and begin gradually. Some people become infected but
don't develop any symptoms and don't feel unwell. Most people (about
80%) recover from the disease without needing special treatment.
Around 1 out of every 6 people who gets COVID-19 becomes seriously
ill and develops difficulty breathing. Older people, and those with
underlying medical problems like high blood pressure, heart problems
or diabetes, are more likely to develop serious illness. People with fever, cough and difficulty breathing should seek medical attention.
User: symptoms?
ChatBot: The most common symptoms of COVID-19 are fever, tiredness, and
dry cough. Some patients may have aches and pains, nasal
congestion, runny nose, sore throat or diarrhea. These symptoms are
usually mild and begin gradually. Some people become infected but
=====
Run TODO Problems Terminal Python Packages Python Console Event Log
PyCharm 2021.1.3 available // Update... (today 09:31) 61.7 CRLF UTF-8 4 spaces Python 3.8 (pythonProject)
```



The screenshot shows the PyCharm IDE with a project named 'pythonProject'. The 'Run' console is active, displaying a chatbot conversation. The user asks 'what are the symptoms of covid', and the chatbot responds with a detailed list of symptoms and recovery information. The user then asks 'what are the symptoms', and the chatbot repeats the same response. Finally, the user asks 'how is it transmitted', and the chatbot provides information on transmission. The interface includes a sidebar with 'Project', 'Structure', and 'Favorites' views, and a bottom status bar showing 'PyCharm 2021.1.3 available // Update... (today 09:31)' and '61:7 CRLF UTF-8 4 spaces Python 3.8 (pythonProject)'.

```
pythonProject - bot.py
Run: bot
User: what are the symptoms of covid
ChatBot: The most common symptoms of COVID-19 are fever, tiredness, and dry cough. Some patients may have aches and pains, nasal congestion, runny nose, sore throat or diarrhea. These symptoms are usually mild and begin gradually. Some people become infected but don't develop any symptoms and don't feel unwell. Most people (about 80%) recover from the disease without needing special treatment. Around 1 out of every 6 people who gets COVID-19 becomes seriously ill and develops difficulty breathing. Older people, and those with underlying medical problems like high blood pressure, heart problems or diabetes, are more likely to develop serious illness. People with fever, cough and difficulty breathing should seek medical attention.
User: what are the symptoms
ChatBot: The most common symptoms of COVID-19 are fever, tiredness, and dry cough. Some patients may have aches and pains, nasal congestion, runny nose, sore throat or diarrhea. These symptoms are usually mild and begin gradually. Some people become infected but don't develop any symptoms and don't feel unwell. Most people (about 80%) recover from the disease without needing special treatment. Around 1 out of every 6 people who gets COVID-19 becomes seriously ill and develops difficulty breathing. Older people, and those with underlying medical problems like high blood pressure, heart problems or diabetes, are more likely to develop serious illness. People with fever, cough and difficulty breathing should seek medical attention.
User: how is it transmitted
ChatBot: The most common symptoms of COVID-19 are fever, tiredness, and dry cough. Some patients may have aches and pains, nasal congestion, runny nose, sore throat or diarrhea. These symptoms are usually mild and begin gradually. Some people become infected but don't develop any symptoms and don't feel unwell. Most people (about 80%) recover from the disease without needing special treatment. Around 1 out of every 6 people who gets COVID-19 becomes seriously ill and develops difficulty breathing. Older people, and those with underlying medical problems like high blood pressure, heart problems or diabetes, are more likely to develop serious illness. People with fever, cough and difficulty breathing should seek medical attention.
PyCharm 2021.1.3 available // Update... (today 09:31) 61:7 CRLF UTF-8 4 spaces Python 3.8 (pythonProject)
```



The screenshot shows the PyCharm IDE with the same project. The 'Run' console continues the chatbot conversation. The user asks 'how is it transmitted', and the chatbot provides information on transmission. The user then asks 'thank you', and the chatbot responds 'Happy to help!'. The user asks 'bye', and the chatbot responds 'See you later'. Finally, the user asks 'bye', and the chatbot responds 'See you later'. The interface includes a sidebar with 'Project', 'Structure', and 'Favorites' views, and a bottom status bar showing 'PyCharm 2021.1.3 available // Update... (today 09:31)' and '61:7 CRLF UTF-8 4 spaces Python 3.8 (pythonProject)'.

```
pythonProject - bot.py
Run: bot
dry cough. Some patients may have aches and pains, nasal congestion, runny nose, sore throat or diarrhea. These symptoms are usually mild and begin gradually. Some people become infected but don't develop any symptoms and don't feel unwell. Most people (about 80%) recover from the disease without needing special treatment. Around 1 out of every 6 people who gets COVID-19 becomes seriously ill and develops difficulty breathing. Older people, and those with underlying medical problems like high blood pressure, heart problems or diabetes, are more likely to develop serious illness. People with fever, cough and difficulty breathing should seek medical attention.
User: how is it transmitted
ChatBot: The most common symptoms of COVID-19 are fever, tiredness, and dry cough. Some patients may have aches and pains, nasal congestion, runny nose, sore throat or diarrhea. These symptoms are usually mild and begin gradually. Some people become infected but don't develop any symptoms and don't feel unwell. Most people (about 80%) recover from the disease without needing special treatment. Around 1 out of every 6 people who gets COVID-19 becomes seriously ill and develops difficulty breathing. Older people, and those with underlying medical problems like high blood pressure, heart problems or diabetes, are more likely to develop serious illness. People with fever, cough and difficulty breathing should seek medical attention.
User: thank you
ChatBot: Happy to help!
User: bye
ChatBot: See you later
User: bye
ChatBot: See you later
PyCharm 2021.1.3 available // Update... (today 09:31) 61:7 CRLF UTF-8 4 spaces Python 3.8 (pythonProject)
```

Thus you can see our chatbot working !!