Report

On

AI Health Care Chatbot for COVID-19

Bachelor In Technology

(Information Technology)

SCHOOL OF CSE (LPU)

PHAGWARA

SUBMITTED TO:

Miss. Nikita Kaushik

Deptt. Of CSE (LPU)

SUBMITTED BY:

Javed Khan 52 (11902959)

Aniket 50 (11905531)

Manikanta 51 (11805409)

Sagar Bhandari 49 (11815751)

LOVELY PROFESSIONAL UNIVERSITY

Phagwara (Punjab)

CONTENTS

Introduction to the Project

A **chatbot** is an artificial intelligence-powered piece of software in a device (Siri, Alexa, Google Assistant etc), application, website or other networks that try to gauge consumer's needs and then assist them to perform a particular task like a commercial transaction, hotel booking, form submission etc . Today almost every company has a chatbot deployed to engage with the users. Some of the ways in which companies are using chatbots are:

- To deliver flight information
- to connect customers and their finances
- As customer support

In this Model we had designed a Bot which can answer all your question related to COVID-19 healthcare.

Overview

1.1

There are broadly two variants of chatbots: **Rule-Based** and **Self-learning.**

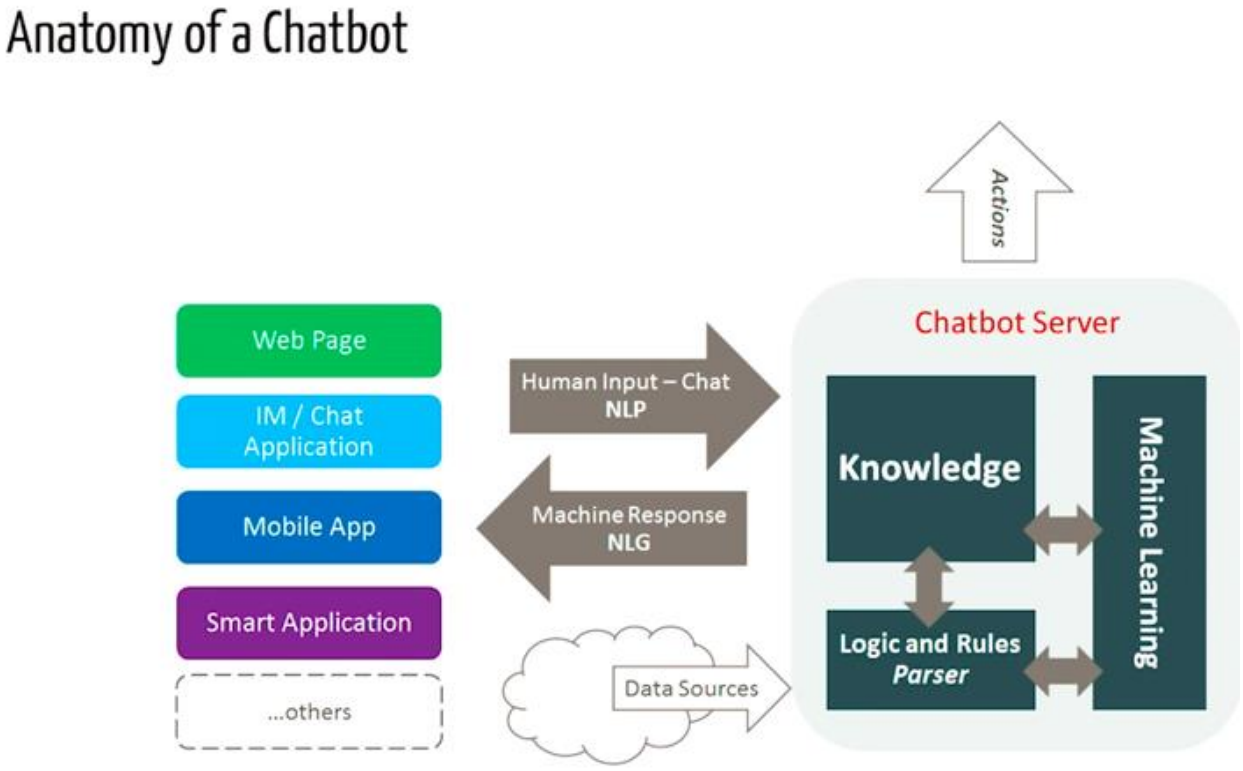
1. In a **Rule-based approach**, a bot answers questions based on some rules on which it is trained on. The rules defined can be very simple to very complex. The bots can handle simple queries but fail to manage complex ones.

2. Self-learning bots are the ones that use some Machine Learning-based approaches and are definitely more efficient than rule-based bots. These bots can be of further two types: **Retrieval Based** or **Generative**

i) In **retrieval-based models**, a chatbot uses some heuristic to select a response from a library of predefined responses. The chatbot uses the message and context of the conversation for selecting the best response from a predefined list of bot messages. The context can include a current position in the dialogue tree, all previous messages in the conversation, previously saved variables (e.g. username). Heuristics for selecting a response can be engineered in many different ways, from rule-based if-else conditional logic to machine learning classifiers.

ii) **Generative** bots can generate the answers and not always replies with one of the answers from a set of answers. This makes them more intelligent as they take word by word from the query and generates the answers.

Anatomy Of Chatbot:



Technology Use And Requirements

Building the Bot

Pre-requisites

Hands-On knowledge of **scikit** library and **NLTK** is assumed. However, if you are new to NLP, you can still read the article and then refer back to resources.

NLP

The field of study that focuses on the interactions between human language and computers is called Natural Language Processing, or NLP for short. It sits at the intersection of computer science, artificial intelligence, and computational linguistics[Wikipedia].NLP is a way for computers to analyze, understand, and derive meaning from human language in a smart and useful way. By utilizing NLP, developers can organize and structure knowledge to perform tasks such as automatic summarization, translation, named entity recognition, relationship extraction, sentiment analysis, speech recognition, and topic segmentation.

NLTK: A Brief Intro

NLTK(Natural Language Toolkit) is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries.

NLTK has been called “a wonderful tool for teaching and working in, computational linguistics using Python,” and “an amazing library to play with natural language.”

Natural Language Processing with Python provides a practical introduction to programming for language processing. I highly recommend this book to people beginning in NLP with Python.

Bag of Words:

After the initial preprocessing phase, we need to transform the text into a meaningful vector (or array) of numbers. The bag-of-words is a representation of text that describes the occurrence of words within a document. It involves two things:

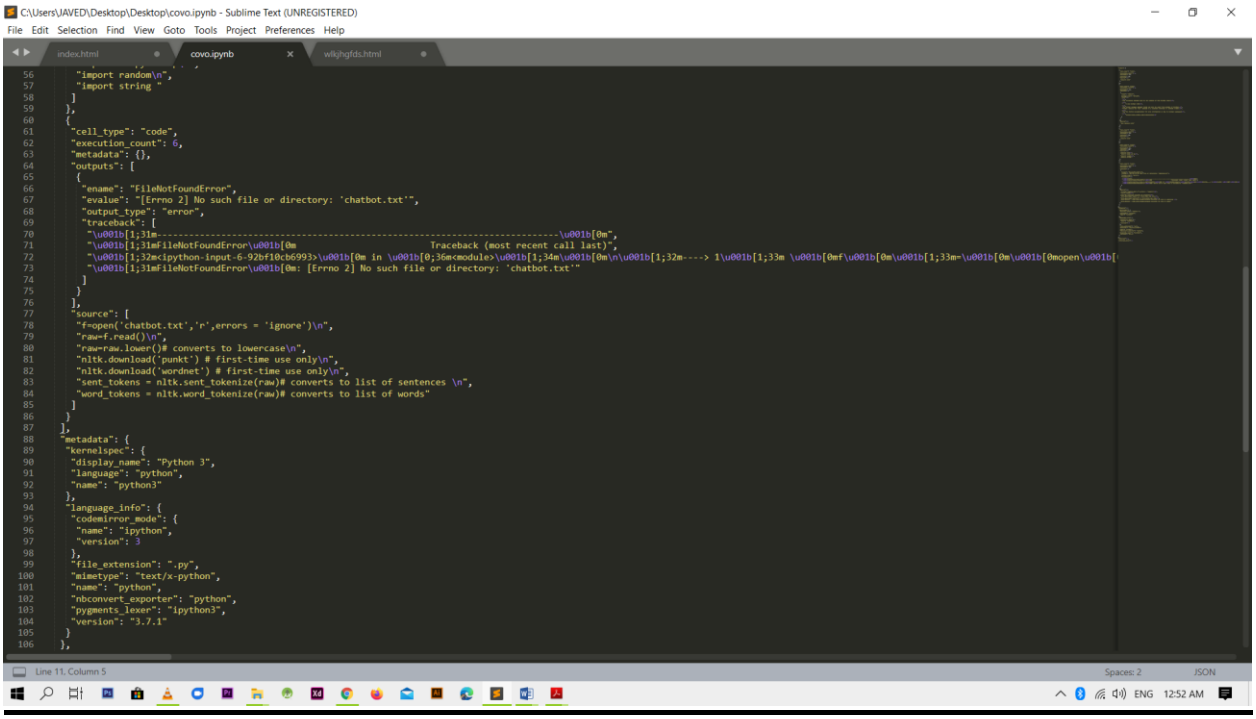
•A vocabulary of known words.

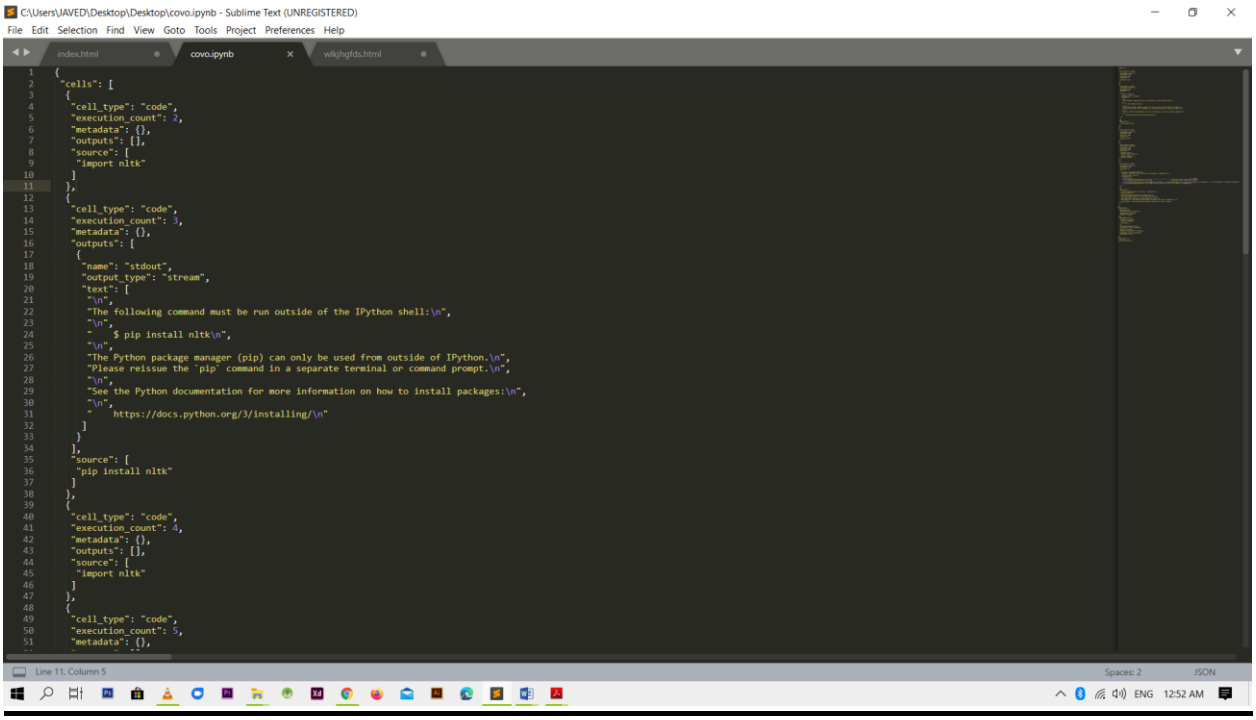
•A measure of the presence of known words.

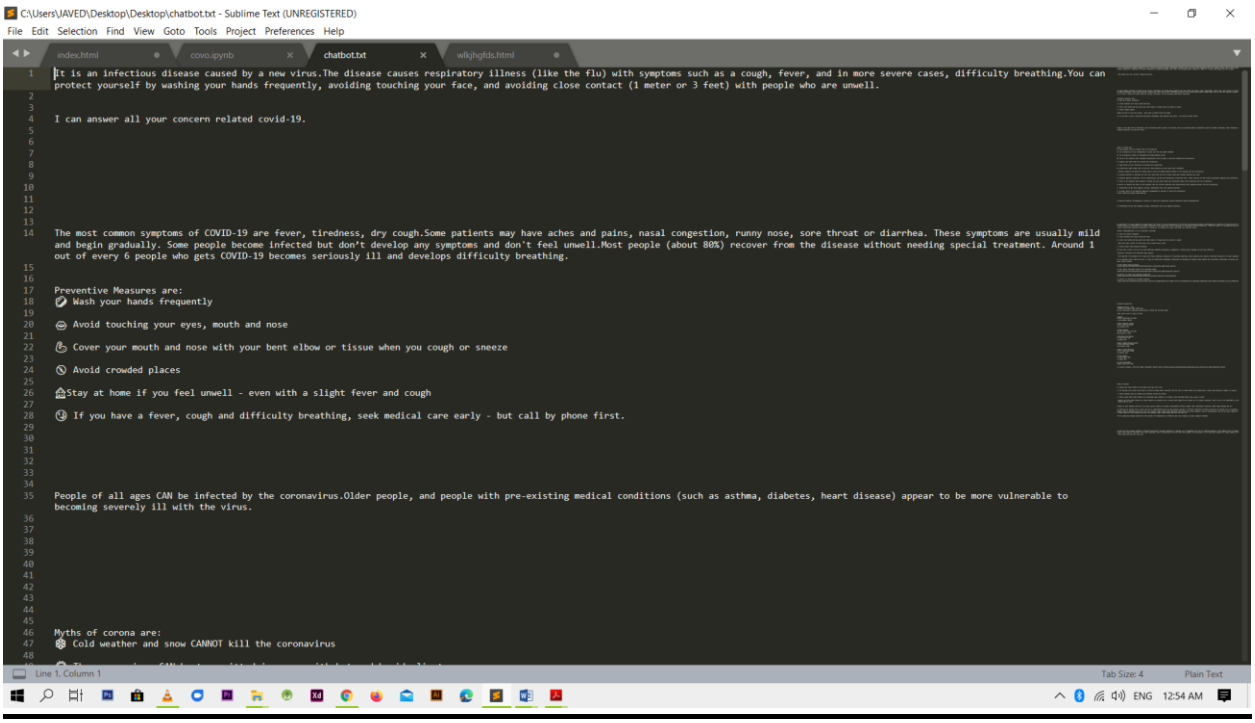
Requirements:

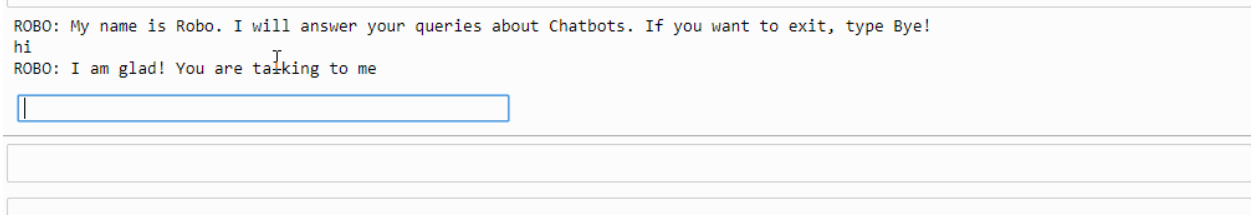
- Microsoft® Windows® 7/8/10 (32- or 64-bit), Ubuntu, Mac OS, Linux
- 1 GB RAM minimum, 2 GB RAM recommended, 2 GB of available disk space
- NLTK Kit
- Python IDE

Screenshots









Scope Of The Project

- This chatbot have a potential to handle out the concern related to covid-19
- Many times patients need a quick piece of advice.It can help you out at such situation.
- The queries would be answered as per the history and current situation.

Algorithms

- **Importing the necessary libraries**

```
import nltk
import numpy as np
import random
import string # to process standard python strings
```

- **Reading in the data**

- We will read in the corpus.txt file and convert the entire corpus into a list of sentences and a list of words for further pre-processing.

- **Pre-processing the raw text**

- We shall now define a function called LemTokens which will take as input the tokens and return normalized tokens.

- **Keyword matching**

- Next, we shall define a function for a greeting by the bot i.e if a user's input is a greeting, the bot shall return a greeting response.

- **Generating Response**

To generate a response from our bot for input questions, the concept of document similarity will be used. So we begin by importing the necessary modules.

- From scikit learn library, import the TFidf vectorizer to convert a collection of raw documents to a matrix of TF-IDF features.

```
from sklearn.feature_extraction.text
import TfidfVectorizer
```

- Also, import cosine similarity module from scikit learn library
  from sklearn.metrics.pairwise import cosine_similarity

  - We define a function **response** which searches the user's utterance for one or more known keywords and returns one of several possible responses. If it doesn't find the input matching any of the keywords, it returns a response:" I am sorry! I don't understand you"

Finally, we will feed the lines that we want our bot to say while starting and ending a conversation depending upon the user's input.

Conclusion:

After careful observation Though it is a very simple bot with hardly any cognitive skills, its a good way to get into NLP and get to know about chatbots.Though 'COVO' responds to user input. We Can implement this module into medical and healthcare unit or portal. It can help you out to handle medical and health concerns**.**

Biblography:

Web links:

- **https://www.stacksoverflow.com**
- **https://www.github.com**
- **https://www.youtube.com**
- **https://geeksforgeeks.com**

Responsibility:

Javed Khan : Complete Coding Part and explanation

Aniket : Corpus Designing

Manikanta : Questonaries and Testing

Sagar Bhandari : Testing & Report