International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 07 | July 2020 www.irjet.net p-ISSN: 2395-0072

DoctorBot - An Informative and Interactive Chatbot for COVID-19

Jeevan Thukrul[1], Aditya Srivastava[2], Gaurav Thakkar[3]

[1-3]*Student, Dept. of Computer Science and Engineering, MIT School of Engineering, MIT ADT University, Pune, Maharashtra, India*

Abstract - *The previous study shows that interactive knowledge is easy to acquire and enhances the understanding level of humans. In the present paper, the research focuses on developing a conversational application, namely Doctorbot on the domain COVID-19. A chatbot is a computer program that suits the requirements of this project. The Coronavirus disease (COVID-19) pandemic caused by the virus SARS-CoV-2 is a highly conversational topic, which makes it a perfect domain for our chatbot.*

The developed chatbot assists in answering user's queries on COVID-19. In this system, the similarity between tokens of the query and the responses from the corpus is taken as the heuristic. The text response generated based on this heuristic value, then the preferred output is then converted speech making it more interactive.

***Key Words*: Heuristic, COVID-19, Pandemic, Corpus, Domain.**

1. INTRODUCTION

The COVID-19 is the most conversational and searched topic in recent times and Chatbots are conversational software that artificially replicates patterns of human interaction. Also, chatbots can be found everywhere, we can find it replacing queries and FAQs in the websites and providing virtual assistance. These features make chatbots useful to spread awareness about COVID-19.

These days chatbots are not only used for general interaction, but they are also built and developed to be domain-specific. The data used in chatbot diversifies with the change in domain. Domain-specific or Closed-domain chatbot makes are designed for a specific area of interest making it thoughtful and relevant. There have been a lot of searches for the pandemic COVID-19 and people coming across false information, so we decided COVID-19 as our domain.

1.1 Types of Chatbot

In our study of chatbots, we found that there are three major types of chatbots namely Rule-based, Retrieval based model, Generative based model [6].

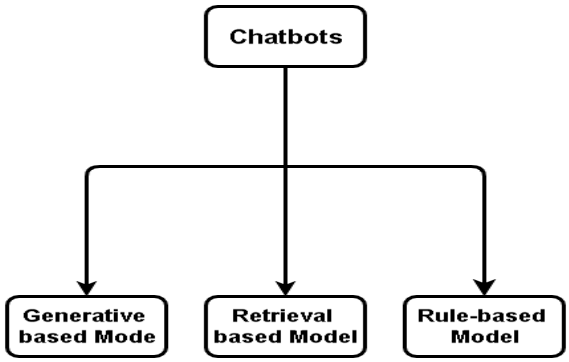

Fig -1: Types of Chatbot.

A) Rule-based Model

In Rule-based Models, a bot answers questions upon a predesigned set of rules. The defined rules can be very simple to very complex. The chatbots designed using this approach can answer a simple and limited number of questions.

B) Retrieval based Model

This nature of the model is such that it uses a limiting condition (heuristic) and selects a response from a set of already defined responses. The bot selects the best response based on the context of the conversation. They are easy to build when compared to generative models.

C) Generative based Model

As the name suggests, these types of bots can generate a response based on the current and the previous experience. Such types of bots are highly advanced and require high computational models and a large amount of data to train.

2. Related Work

The chatbot is designed for textual communication for the educational domain. The chatbot uses vectorization on the response of the user to extract the features. Based on these extracted features Random forest algorithm is implemented to search the responses in the dataset [5].

The Chatbots are a computer program, built to fulfill a particular purpose. A chatbot is an idea that is developed using various machine learning and deep learning techniques to make it conversational and serve the purpose. It serves as a comparison between early chatbot examples like ELIZA, ALICE, PARRY, JABBER WACKY on their architecture, and various other parameters. [6]

The built system is a web-based chatbot developed to share university information with the users. The chatbot uses keywords present in the user's message and executes the SQL query to find the responses which match the array of keywords. [7]

A virtual assistant built as an Android application assists people in stress management. The chatbot system uses the Encoder-Decoder Sequence model of the RNN (Recurrent Neural Networks). A large amount of dataset is required to implement this model. [8]

The chatbot JARO serves the purpose to automate the interview process. The chatbot uses multiple features like sentiment analysis, Natural language processing, and automatic question generation to score the candidate. The core part of the chatbot works on keyword matching, NLP, and Sentiment analysis. A report is generated based on the recorded responses. [9]

A counseling application is developed in the form of a chatbot to assist the individuals. It Implements various Emotion recognition to check the emotions of the user based on the history of responses. The response is generated using the Recurrent Neural Network (RNN). Each response is encoded based on the previous responses. [10]

3. METHODOLOGY

Doctorbot is implemented using Retrieval based approach, by employing Natural language processing. The block diagram of the proposed system is elaborated in Fig-2. The chatbot is developed to return the most similar data from the corpus, based on the query.

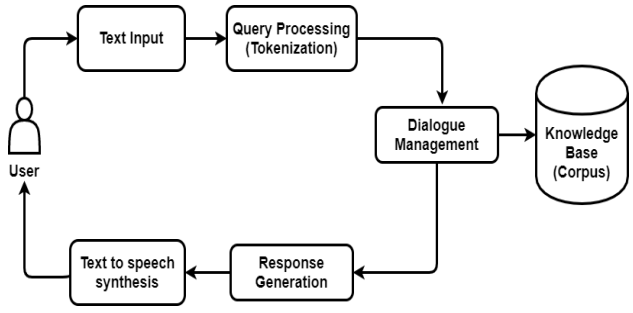

Fig-2: Block Diagram of Proposed System

[1] NLP

Natural Language Processing (NLP) is a branch of AI, which deals with the interaction among machines and human spoken languages. NLP helps a computer to analyze and understand the user's input. Using NLP computers can perform various tasks like knowledge extraction, sentiment analysis, speech recognition, fake news detection. Natural Language Processing was developed based on a set of mathematical rules which in turn developed with Machine Learning algorithms for language processing.

[2] Data Preprocessing

Data preprocessing is the most essential step when it comes to machine learning. Machines don't understand the data as humans, and that's the reason data must be preprocessed. Machine learning algorithms need numbers to work upon and so we use numerical feature vectors. Basic Data preprocessing includes:

Letter Case: The complete corpus of predefined responses or user responses must be in a single case, preferably lower case. This process helps the algorithm used in the bot not treat the same words in both cases as different and maintain uniformity.

Tokenization: Tokenization is what is used to describe the process of converting the normal text-based strings into a list of tokens that can be further processed. Sentence tokenizer and Word tokenizer are used to make a list of tokens from words and sentences respectively.

Removing Noise: Everything other than the standard number or letter should be removed.

Removing Common words: The common words that are of negligible value and help select and match the user needs have to be excluded. These words are also called stop words since they are the words where the machine literally stops analyzing.
Stemming: The process which reduces derived words to their stem and root form. For example, if we were to stem the following words: “Writes”, “Writing”, “Written” and "Written", it would result in a single word “write”.

Lemmatization: Lemmatization is similar to stemming. The difference between stemming and lemmatization is that stemming can make words in their root word, whereas lemmas are the words that are actually present in the response. So, your root stem, meaning the word may not be something that can always be found in a dictionary, but on the other hand, lemma can be found.

[3] User Request Analysis

In this step, the Chatbot analyses the queries requested by the user and recognizes the user intent to extract relevant data objects. This is considered to be the most crucial step of the entire working of the application because without intent the application will not return a righteous response.

The word embedding based on the frequency is used to predict the appropriate response [6]. The TfidfVectorizer is used to extract the features from the user's response [5].

[4] Returning the Response

Once the user request is analyzed, the application can now respond to the user's query. In this particular system, the answer can be:

1) A Predefined response, encoded in the program.

2) A text retrieved from the corpus (Knowledge-base).

Once the intent of the query is correctly interpreted, the application finds it easy to respond based on various machine learning approaches. The application Doctorbot uses cosine similarity to infer the most similar sentence from the corpus as the response.

The features extracted based on the frequency using TfidfVectorizer are compared using the cosine similarity. Therefore, in this system the heuristic used in the system is frequency. The most similar sentence is selected as an output from the corpus, based on the calculations from TfidfVectorizer and cosine similarity.

4. MATHEMATICAL MODEL

[1] TfidfVectorizer

Tf means term-frequency while tf–idf stands for term-frequency times inverse document-frequency [2], [4]:

$$\text{tf-idf}(t,d) = \text{tf}(t,d) \times \text{idf}(t) \quad (1)$$

$$\text{idf}(t) = \log n + 1\text{df}(d,t) + 1 + 1 \quad (2)$$

Here, in equation (2),
n is the complete number of responses in the corpus, df(t) is the number of responses in the corpus that contain term t.
Then tf-idf vectors are then normalized by the Euclidean norm:

$$V\,norm = \frac{v}{\sqrt{v_1^2 + v_2^2 + \ldots + v_n^2}}$$

[2] Cosine similarity

Cosine similarity is a metric measure used to check the similarity in the documents, irrespective of their size [3].

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}$$

In a document, the cosine similarity measures the orientation (the angle) of the documents instead of magnitude, corresponding to a particular term.

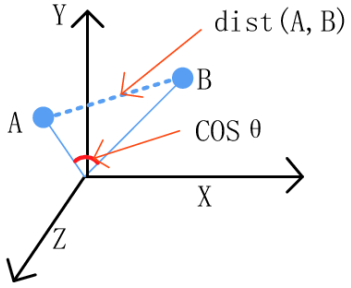

Fig-3: Difference in Euclidean distance and cosine similarity

Smaller the angle between the vectors, the higher the similarity.

5. RESULTS

The system developed serves the purpose. It properly extracts the features using the TF-IDF vectorizer and returns the most relevant response based on the cosine similarity index. Though, the responses have a limitation corresponding to the provided dataset or corpus.

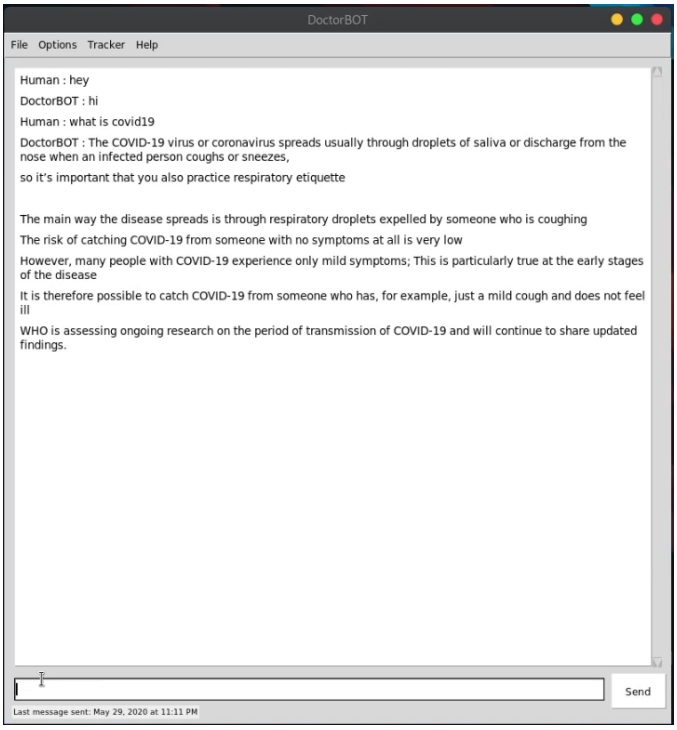

Fig-4: The Chatbot with GUI

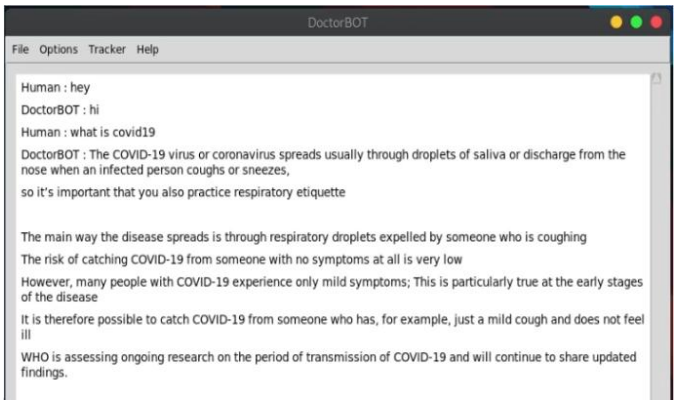

Fig-5: The response based on the cosine similarity index

6. CONCLUSIONS

The project developed and researched is an interactive chatbot developed using Artificial Intelligence. The domain was considered based on the hot topic COVID-19. The training data to develop this chatbot was devised from the data available on the World Health Organisation (WHO) website. In this system, an approach based on Retrieval Algorithms is implemented to get the response that is most relevant. To make the system more responsive, the response given by the system has a feature of Text to speech conversion.

The Doctorbot based on the medical domain has a large scope for this project, which will precisely answer all the major queries of the users.

REFERENCES

[1] https://www.who.int/

[2] https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html

[3] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.pairwise.cosine_similarity.html

[4] https://www.worldometers.info/coronavirus/

[5] Anupam Mondal, Monalisa Dey, Dipankar Das, Sachit Nagpal, Kevin Garda "Chatbot: An automated conversation system for the educational domain." IEEE, 2018.

[6] K. Jwala, G.N.V.G Sirisha, G.V. Padma Raju "Developing a Chatbot using Machine Learning" International Journal of Recent Technology and Engineering (IJRTE) Volume-8 Issue-1S3, June 2019.

[7] Neelkumar P. Patel, Devangi R. Parikh, Prof. Darshan A. Patel, Prof. Ronak R. Patel "AI and Web-Based Human-Like Interactive University Chatbot (UNIBOT)" IEEE, ICECA 2019.

[8] Aafiya Shaikh, Dipti More, Ruchika Puttoo, Sayli Shrivastav, Swati Shinde "A Survey Paper on Chatbots" International Research Journal of Engineering and Technology (IRJET), Volume: 06 Issue: 04, Apr 2019.

[9] Jitendra Purohit, Aditya Bagwe, Rishabh Mehta, Ojaswini Mangaonkar, Elizabeth George "Natural Language Processing based Jaro-The Interviewing Chatbot" Third International Conference on Computing Methodologies and Communication, IEEE, 2019.

[10] Kyo-Joong Oh, DongKun Lee, ByungSoo Ko, Ho-Jin Choi "A Chatbot for Psychiatric Counseling in Mental Healthcare Service Based on Emotional Dialogue Analysis and Sentence Generation", 18th International Conference on Mobile Data Management, IEEE, 2017.