

ARTICLE

CO-BOT: An Intelligent Technique for Designing a Chatbot for Initial COVID-19 Test

Projit Das¹ Ira Nath¹ Avishek Banerjee² Laxminarayan Sahoo^{3*}

1. Department of CSE, JIS College of Engineering, Kalyani, 741235, India
2. Department of CSBS, Asansol Engineering College, Asansol, 713305, India
3. Department of Computer and Information Science, Raiganj University, Raiganj, 733134, India

ARTICLE INFO

Article history

Received: 28 November 2022

Revised: 15 December 2022

Accepted: 19 December 2022

Published Online: 28 December 2022

Keywords:

Chatbot

COVID

Cloud

ABSTRACT

The coronavirus (nCoV-19), which was discovered, has now spread around the world. However, managing the flow of a large number of cases has proven to be a significant issue for hospitals or healthcare professionals. It is becoming increasingly challenging to speak with a medical expert after the epidemic's initial wave has passed, particularly in rural areas. Thus, it becomes clear that a Chatbot that is well-designed and implemented can assist patients who are located far away by advocating preventive actions, and viral updates in various cities, and minimising the psychological harm brought on by dread. In this study, a sophisticated Chabot's design for diagnosing individuals who have been exposed to COVID-19 is presented, along with recommendations for immediate safety measures. Additionally, when symptoms grow serious, this virtual assistant makes contact with specialised medical professionals.

1. Introduction

To stop the country from seeing an increase in COVID-19 instances, numerous research projects are now being conducted. Before the successful development of these kits, our nation imported medical supplies, including PPE (Personal Protection Kits) and masks, from abroad. Our nation has taken steps to raise awareness of the disease as

well as initiatives to combat it. The news and media play an important role in fostering this knowledge by educating the public about the preventive steps that can shield them against infection. Keeping a virus from spreading can be made much harder if people are more aware of the need to take all precautions.

COVID-19 has been quickly spreading around the world. Many people have started to show various viral

*Corresponding Author:

Laxminarayan Sahoo,

Department of Computer and Information Science, Raiganj University, Raiganj, 733134, India

Email: lxshoo@gmail.com

DOI: <https://doi.org/10.30564/jcsr.v4i4.5287>

Copyright © 2022 by the author(s). Published by Bilingual Publishing Co. This is an open access article under the Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License. (<https://creativecommons.org/licenses/by-nc/4.0/>).

symptoms. When persons with impaired immune systems, the elderly, or those with chronic conditions experience COVID-19 symptoms, it might cause considerable complications^[1-4]. Because of the burden, this put on hospitals and medical call centres, people began to become enraged with the COVID-19 virus, especially after multiple false reports about it were spread on social media. For instance, in March 2020, the developing coronavirus resulted in more than 1,400,000 calls to Saudi Arabia's Ministry of Health in a single month^[5]. AapkaChikitsak, an Indian Chatbot^[6], and ANA, a Brazilian Portuguese Chatbot^[7], both of which were developed to counteract fake news and misinformation and provide trustworthy information regarding COVID-19. Furthermore, several recently released research studies have chosen to use chatbot technology, which is useful in providing telehealth services in the COVID-19 pandemic scenario^[8,9] or for numerous goals connected to the healthcare industry, such as medical diagnosis, treatment of drugs, and pathology queries, or even for general medical awareness purposes^[10,11].

Using text or voice, a chatbot is computer software that converses with users in normal language, interprets their intentions, and then responds following the organization's business policies and data. ELIZA^[12] was among the initial chatbots. It served as an early test case for the Turing Test, which examines whether a machine's intellect can be compared to or is indistinguishable from that of a person. Early consumers felt they were speaking with a person who understood their input since the computer utilised "pattern matching" and replacement methods to offer prepared responses. The software's scripts restricted its functionality. In the middle of the 1960s, Joseph Weizenbaum, a researcher at MIT's Artificial Intelligence Laboratory, developed a pioneering natural language processing method. It was supposedly created to demonstrate how tenuous ties existed between humans and computers at the time. On the other hand, when it was put on computers, people found it quite entertaining. Chatbots have a wide range of uses in the modern world, including online commerce, ticketing, news reporting, meal ordering, etc.

More safety and caution regarding this condition are provided by this chatbot system. The AWS platform, which enables users to forego expert advice, can be used to develop this system. It is designed to determine the user's ailment and deliver pertinent information about the disease. With enhanced access to information regarding diseases, this method is designed to be cost-effective. Only when a Chatbot can diagnose any illness and give consumers the information they need will it be of benefit to them. The suggested system uses a conversational agent to interact with individuals and learn more about

their medical issues in order to get an accurate diagnosis. Today's digital environment allows for very high amounts of information interchange per minute, enabling people to conduct business from afar without having to be physically there. Using Human Machine Interaction (HMI) technology, which cut out humans from the process, is used in many services and interaction settings. Despite recent technological advancements, humans will still be an integral part of the contact loop for a very long time. They mostly act as "answers" machines, which is their fundamental issue; they are unable to reliably and effectively react to human inquiries. We propose a novel HMI scheme to address this issue that is based on SPN discourse rather than providing responses to inquiries and is capable of improved communication and engagement with human users.

The objective of the proposed work

The following are the observations of my proposed work:

- (i) To determine the possible outcome and analysis COVID report based on some given health conditions.
- (ii) This chatbot will store all the responses given by the users and then after analysing the data available doctors or executives call them and discuss their situations.
- (iii) The chatbot is used by users to comprehend the facts and news surrounding the sickness, to learn how to protect themselves from the coronavirus, and to help stop the spread of the illness.
- (iv) This chatbot gives us information about the comparative situation of COVID-19 trends in different cities.

2. Related Work

There have been several text-based human-computer interaction systems created, such as ELIZA^[12], which mimics a psychiatrist, and PARRY^[13], which indicates the thoughts of a patient who is paranoid. In two different trials, Rajj et al.^[14] contrasted virtual human interactions with real human interactions in the context of medical consultation. Their findings demonstrate parallels between real-world and virtual interactions. An article by Fadhils et al.^[15] demonstrates the use of intelligent conversational systems to communicate with elderly people in order to gather data and conduct ongoing health condition monitoring, particularly after hospital release. A medical recommendation system particularly created to engage with users and take on the role of a doctor is presented by Amato et al.^[16,17]. Pharmabot is a paediatric generic medicine consultant chatbot that Comendador et al. introduce. It is intended to refer to and give relevant data on generic

drugs for kids.

A chatbot that is a personalised conversational assistant was built by Poongothai. M. et al. ^[18] to aid students in asking inquiries and resolving problems regarding current and completed Internet of Things projects. The advantages of this system include a few characteristics including auto-scaling, real-time notifications, and quick response. Later, their laboratory system was upgraded to include this chatbot.

Ali Parsa founded Babylon Health ^[19] in 2013, which is a provider of value-based healthcare, that combines an AI-powered stage with virtual clinical procedures for sick persons. Through its website and mobile application, patients are connected with healthcare providers. Users can email queries or photographs to the medical specialist in a manner similar to text messaging. Patients can also video chat with doctors to seek guidance for their health problems. Additionally, users can get referrals to doctors, have their prescriptions for pharmaceuticals delivered to them or forwarded to a pharmacy, and consult with therapists to discuss problems like depression and anxiety. When a physical examination is required, users in London can book health exams with a select group of institutions; nurse visits, however, are only offered at one location.

According to its website, this healthcare chatbot ^[20] attempts “to assist doctors in their daily work”. It functions as an assistant that gives medical users precise data through chat by offering helpful advice and keeping track of a woman’s health while she is nursing. Additionally, it assists healthcare professionals in advising women on the appropriate drugs to take while they are breastfeeding. This can be incredibly beneficial for people to search for quick, accurate information anytime it’s needed, saving time by having all the information at their fingertips rather than having to spend time looking it up.

Florence ^[21] chatbot, which functions on Facebook Messenger, Skype, or Kik, is essentially a “personal nurse”. She can remind people to take their medication, which could be a useful feature for elderly patients. You simply need to enter the drug’s name, how frequently you need to take it, and when in the chat window. Florence then casually reminds you to take the prescription every time you need to. The user’s health, including things like body weight, mood, and menstruation, may also be monitored by Florence, who will help them succeed in their goals. The chatbot can also assist you in finding the nearest pharmacy or medical facility if necessary.

To enable objective data reception and transmission in real-time, “Smart Wireless Interactive Healthcare Systems” (SWITCHes) ^[22] are being developed. In addition to engaging users with customised feedback in an interactive

manner through a health chatbot powered by artificial intelligence, the SWITCHes app can also connect users with healthcare professionals who can offer them more precise medical advice depending on the customer’s data collected from the app and auxiliary data from medical devices. SWITCH’s technology differs from researcher-controlled and commercial apps because of an integrated health chatbot. Users can converse with the health chatbot and receive information in real-time, or they can heed the bot’s counsel and adhere to its recommendations for healthy living, which may include diet and exercise regimens. The AI-powered health chatbot may provide real-time information as well as assist users in keeping track of their health issues and staying motivated by providing inspiration and reminders. The healthcare practitioner can give the user more precise medical recommendations based on the user’s data collected from the SWITCHes app and the supporting data from medical equipment. A health chatbot powered by artificial intelligence can also communicate with SWITCHes app users to provide personalised feedback. SWITCHes distinguishes itself from researcher-controlled and for-profit apps with its built-in health chatbot.

With the aid of Tidio’s ^[23] conversational AI chatbot, you can improve your customer service and increase sales. You can design your AI bot exactly how you want because it is simple to use. You may also use a visual builder interface to watch how it develops. Your bot will use NLP technology to interact with and serve your customers more effectively. And you’ll be accessible to your clients around the clock to ensure that you don’t miss any chances to close deals.

3. Proposed Method

Conversational voice and text interfaces can be created using Amazon Lex. Your applications can incorporate chatbots that you create with Amazon Lex. Internally, Amazon Lex uses the same deep learning platform as Amazon Alexa. Amazon Lex’s internal technology combines natural language understanding (NLU) and automatic speech recognition (ASR) to comprehend the intent of the text and convert speech to text. As this course progresses, we’ll go into more detail on these subjects. Essentially, Amazon Lex frees you from the difficulties of speech recognition and natural language understanding so that you can quickly create really engaging chatbots. A budget-friendly option is Amazon Lex. Like the majority of AWS services, Amazon Lex offers pay-as-you-go pricing and has no upfront expenses. Utilizing a mixture of aliases and versioning, Amazon Lex offers deployment functionalities that let you rapidly roll out your conversational interfaces across numerous environments with-

out any difficulties. You may scale up with Amazon Lex without worrying about bandwidth usage because Amazon Lex does not place restrictions on it. Finally, there is seamless integration between Amazon Lex and a number of other AWS services. Natural language understanding and automatic speech recognition, or ASR, for speech-to-text conversion offer sophisticated ways to transform your voice or text commands into commands that can be carried out. Intents, utterances, slot kinds, slots, and channels are just a few examples of the many child configuration pieces that make up a bot. As we go on, we'll discuss each of these in more detail. Within Amazon Lex, the bot itself serves as the building and deployment unit. Developers are able to create and deploy a variety of bots, each with its own unique set of capabilities. Several child configuration pieces, including intents, utterances, slot types, slots, and channels, to mention a few, make up a bot. In the coming sections, we'll go through each of these in detail. Within Amazon Lex, the bot itself serves as the unit of construction and deployment. The ability to create and deploy numerous bots, each with a unique set of abilities and behaviours, is available to developers. The results that the bot might take are represented by intent. For instance, scheduling an appointment, booking a flight, or reserving lodging are all examples of intentions. A term that accurately conveys what the objective achieves, Utterances. The user says or enters one or more phrases that trigger the purpose, and fulfilment procedure- how the intent was carried out or fulfilled. Aside from the built-in intentions that Amazon Lex offers, you can also use your own custom intents. For each goal, the user may need to give more qualities, also called slots, in order to get the desired result.

The working method of this chatbot is that the user first gives some query-related statement to the chatbot, and the chatbot will check the spelling with its stored intents. After that, if both statements are the same, the chatbot will reply, and then our chatbot will ask for some personal information from the user. The next user asks questions as per their demand. If they want to check COVID status or COVID-related basic questions, or if COVID updates in different cities, our chatbot will reply as per stored value in the database. In the end, the chatbot will show the thank you, and if the user gives some wrong inputs, it will show an error response. Here in Figure 1, we briefly describe the working flow diagram of our chatbot.

Figure 1 describes when a user or patient gives some input or comments as per their needs, then CO-BOT will start replying. Before that, CO-BOT will check the input coming from the user with its database. If it matches the database, it will give a suitable response. CO-BOT will

save all the information given by a user for further response.

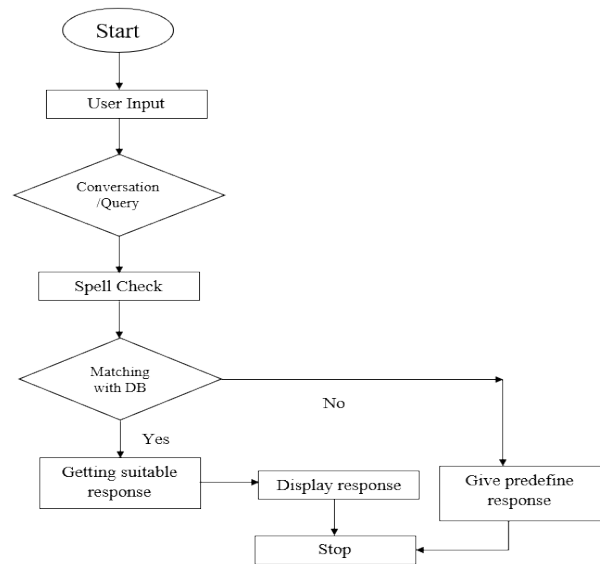


Figure 1. Flow Diagram of COVID Chatbot

In this project, we used Amazon Lex API to create the sample utterances and intent from a chatbot, then connected this Lex with a lambda function and did the validation also. Then create IAM and Cognito for authentication and connect these with S3. Also, connect S3 with a lambda function for static web hosting. Here, by using Figure 2, we have given a clear architecture of our project.

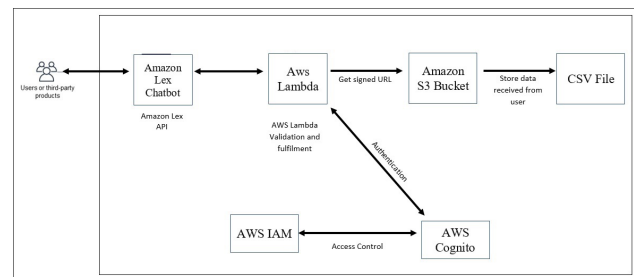


Figure 2. System Architecture of Chatbot

Figure 2 briefly describes the actual working system of CO-BOT. The Amazon Lex API is used to create a bot and configure it with one or more intents that you want to support. Configure the CO-BOT as per the basis of the patient goal, and engages in conversation with the user or patient to elicit information. Next, lex is connected with lambda which allows us to add custom logic to AWS resources such as Amazon SE bucket and also allows quick calculation or processing as the input to other functions. When we connect projects with Amazon S3 organizer, and manage our data in such a way that supports specific use cases, enables cost efficiencies, enforces security and meets compliance requirements. When we connect with

a CSV file, it will store all the responses of a patient or user and use them for future reference. Here, using temporary security credentials, Amazon IAM roles offer a mechanism to access AWS. A role is not associated with a specific user or group, but rather with a set of rights for requesting AWS services. With Amazon Cognito, adding user sign-up and authentication to your mobile and online apps is a snap. With the help of Amazon Cognito, you can use AWS or any service behind Amazon API Gateway to get to the backend resources of your app. Amazon Cognito also gives you temporary security credentials.

3.1 Amazon Lex Component Design

In Amazon Lex, a bot is the main kind of resource. Intent, slot type, alias, and bot channel association are the additional resource kinds in Amazon Lex. You can develop a bot using the model-building API or the Amazon Lex console. You can create a production-ready bot for your application using the graphical user interface offered by the console. You can develop a bot using your own custom software using the model-building API through the AWS CLI if you'd like. You can either integrate a bot into your own application or deploy it on one of the supported platforms after it has been created. When a user interacts with the bot, the client application uses the Amazon Lex runtime API to make queries to the bot. For instance, your client uses one of the runtime API calls to transmit the user's input to Amazon Lex when they say, "I want to buy pizza." Users can enter information orally or in text form. Sample utterances are structured as:

```
<what is the condition in {city}>
<tell me {city} affection>
<Want to know about covid>
```

3.2 Lambda Function in AWS

Lambda functions are supported by Amazon Lex so that the bot can get code hooks. These features can accomplish a variety of tasks, including enhancing user engagement with the bot by utilising existing knowledge, validating the user-provided input data, and carrying out the user's goal.

Figure 3 is a snapshot of our lambda function. By using this lambda function, we have done all the condition checking like whether the patient has a fever or not, breathing issues or not, checking 10-digit contact details, etc. and also connected a CSV file to store all the details of the patient's physical health and contact details for future reference. For searching, add more queries, and to control these queries, add some functions that check if certain conditions are met.

```

# LAMBDA FUNCTION CONNECTED WITH A 'CSV' FILE
import boto3
import csv

# call s3 bucket
s3 = boto3.resource('s3')
BUCKET_NAME = "new-covidbot-bucket-user-data"
bucket = S3.Bucket(BUCKET_NAME) # enter your bucket name, e.g 'data'
# key path, e.g. 'customer_profile/Reddit_Historical_Data.csv'
key = "user_data.csv"

def positive():
    return "Thank you for providing the answers!! You might have covid, no need to worry our executive will be in touch"
def negative():
    return "Thank you for providing the answers!! It seems you do not have Covid, still our executive will contact you on your"

def append_data_to_s3(fname, lname, con_no, report):
    # download s3 csv file to lambda tmp folder
    local_file_name = "/tmp/user_data.csv" #
    s3.Bucket(BUCKET_NAME).download_file(key, local_file_name)
    # list you want to append
    lists = [fname, lname, con_no, report]
    print(lists)
    # write the data into '/tmp' folder
    with open('/tmp/user_data.csv', 'r') as infile:
        reader = list(csv.reader(infile))
        reader = reader[1:-1] # the data is ascending order in file
        reader.insert(0, lists)
    with open('/tmp/user_data.csv', 'w', newline='') as outfile:
        writer = csv.writer(outfile)
        for line in reversed(reader): # reverse order
            writer.writerow(line)
    # upload file from tmp to s3 key
    bucket.upload_file('/tmp/user_data.csv', key)

```

Figure 3. Sample Lambda function

3.3 Amazon Cognito

AWS's Cognito service makes it simple to integrate user management into online and mobile applications. It works with SAML 2.0 and provisions social identity providers like Facebook, Google, and business identity providers. This is an effective service and hard to grasp at first.

The fact that there are two main components of Amazon Cognito is one of the things that causes the most confusion:

- 1) User Pools for Amazon Cognito
- 2) Amazon Cognito Identity Pools, second (aka Federated Identities)

3.3.1 Cognito User Pool

This means an anonymous user of our application (e.g. a mobile or a Single Page Application) can fill out a registration form and then become a registered user. The chosen credentials (i.e., username and password) will be safely stored in the Cognito User Pool.

In this case, Amazon Cognito acts as an Identity Provider (IdP). When this registered user wants to log in, the User Pool will be used as the source of truth to assess the authenticity of provided credentials; if valid, a JSON Web Token (JWT) will be returned.

3.3.2 Cognito Identity Pool

Usually, REST APIs are protected through the use of a token – e.g., a JSON Web Token (JWT) – and that's why Amazon API Gateway with the help of Cognito User Pool, supports this scenario natively. Alas, the vast majority of AWS resources don't support a JWT as a means of authentication! For instance, if our application reads order item 42 directly from DynamoDB, we need an IAM Role that has permission to read data from the Orders table.

3.3.3 AWS Credentials

```
// Initialize the Amazon Cognito credentials provider
```

```
CognitoCachingCredentialsProvidercredentialsProvider =
new CognitoCachingCredentialsProvider(getApplication-
Context(),
“us-east-1:77511a8f-cb2d-488f-8cd5-233e16e3137c”, //
Identity pool ID
Regions.US_EAST_1 // Region
);
```

3.4 Amazon S3

In order to store and access any amount of data or information from anywhere online, Amazon S3 [24] (Simple Storage Service) provides object storage. It was designed for developers to help with web-scale computing and offers 99.999999999 percent durability and 99.99 percent availability of items. Data file up to 5 terabytes in size can also be stored on it.

Figure 4 describes the web hosting of our project. Using this feature, we can create a bucket and upload a .html file for accessing the chatbot. After enabling static web hosting we can use that chatbot virtually by using a single link.

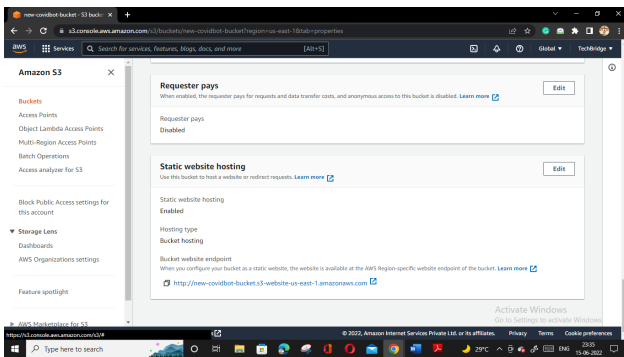


Figure 4. Static Web Hosting from S3

3.5 Data Storing Process in Amazon s3

Objects are used to store data on Amazon S3. With this method, cloud storage is greatly scalable. Different physical disc drives dispersed around a data centre can house various objects. Amazon datacentres use specialised hardware, software, and distributed file systems to offer great scalability. The block storage method implements features like redundancy and versioning. By default, when a file is kept in S3 as an item, it is simultaneously stored in several locations (such as discs, data centres, or availability zones). Control hash sums are frequently checked as part of the S3 service’s data consistency verification. If data corruption is found, redundant data is used to recover the object. S3 buckets are used to store objects. Objects in

S3 storage can, by default, be viewed and handled through the web interface [24].

Object URL for error.html

https://webbucket07.s3.amazonaws.com/error.html

Object URL for index.html

https://webbucket07.s3.amazonaws.com/index.html

Bucket Policy

```
{
  “Version”: “2012-10-17”,
  “Statement”: [
    {
      “Sid”: “PublicReadGetObject”,
      “Effect”: “Allow”,
      “Principal”: “*”,
      “Action”: “s3:GetObject”,
      “Resource”: “arn:aws:s3:::webbucket07/*”
    }
  ]
}
```

Figure 5 shows the bucket to upload HTML files like index.html for the front-end interface of the chatbot and error.html represent the error that occurs interface of the chatbot. From this terminal, we can also enable the web hosting part that helps to access our chatbot project.

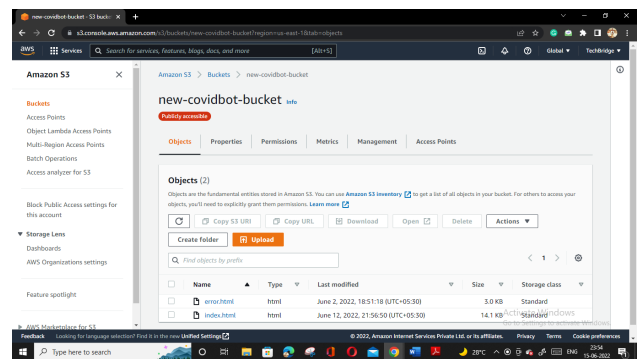


Figure 5. Upload the HTML file to the bucket

For storing data, we have used a CSV file which stores all the records of COVID feedback. Patients will give their health situation by selecting the different conditions from the auto-generated options. On the basis of given feedback, the bot will analyse if COVID has been or not. Then this CSV file will store the name of the patient, contact number, and status of COVID. Later on, the patient will get a call from an executive and discuss COVID over the phone.

Figure 6 shows the CSV which stores all the records of patients and from this given contact number we can contact that patient in the future as per need.

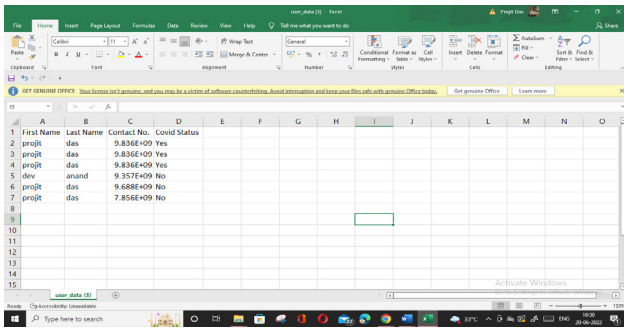


Figure 6. Store user COVID report data

3.6 Valuation of Dataset

Any chatbot should appear normal when reacting to user input, and it must have a clean dataset and long-lasting backend logic for result production. We have created a ten-question COVID-19 basic symptomatic questionnaire at the Telemedicine department at the University of Camerino. Simple Yes/No questions make up the entirety of this survey. The bot delivers a score of one if the user selected “Yes”, and a score of zero if the user selected “No”. The chatbot anticipates using AIML logic to respond to human input and maintain input that could feed the machine.

Table 1 lists the early symptoms that can be used by the chatbot to determine whether or not a person is infected.

Table 1. Characteristic analysis dataset

No.	Questions	Type
1	Confirm your age.	Enter an int value
2	Do you have a fever?	Yes/No
3	Do you have a cough?	Yes/No
4	Do you have a sore throat	Yes/No
5	Do you have chest congestion or a runny nose?	Yes/No
6	Do you facing any breathing issues	Yes/No
7	Do you have diabetics?	Yes/No
8	Do you have Hypertension?	Yes/No
9	Do you have lung disease?	Yes/No
10	Do you have heart disease?	Yes/No
11	Have you travelled in the past 14 days to any states?	Yes/No
12	Which city do you want to know?	Select any city

4. Result and Discussion

Initially, the chatbot successfully provided the right output in about 40% of cases. However, as the training data improved with continuous interaction with the chatbot, this accuracy improved to 70%. For queries unrelated to the intents described in the chatbot, a generalized response is generated, and such questions are logged to be

checked later and included in the datasets. The Amazon S3 connected to the chatbot has successfully monitored the traffic to various sections of the chatbot by capturing the general purpose of the user. This information can be further used to analyse the kinds of people visiting the site and their purpose. Hence, more emphasis can be put on such for performance testing. Two parameters were evaluated: answer delivery delay and question matching accuracy. 1) Answer Delivery Delay: It is defined as the time from the question sent by the user until he/she receives the answer. This latency was constantly under 3 seconds, with an average of 1.76 seconds. 2) Matching Accuracy: In this, we check whether the chatbots work even when the sentences are mistyped, and the proposed system has successfully matched the sentences to the predefined utterances in 70% of the cases.

When a user gives some statements to the chatbot, it will then ask for some information from the user. Here users have to put some basic information to us for validation. The user must respond as required, after which the chatbot will ask for the user’s contact information and confirm whether or not the user has COVID. Snapshots of this feature are given below in following Figures 7 and 8. Getting all the information regarding COVID will be stored in a CSV file from where an executive or doctor can call the patient.

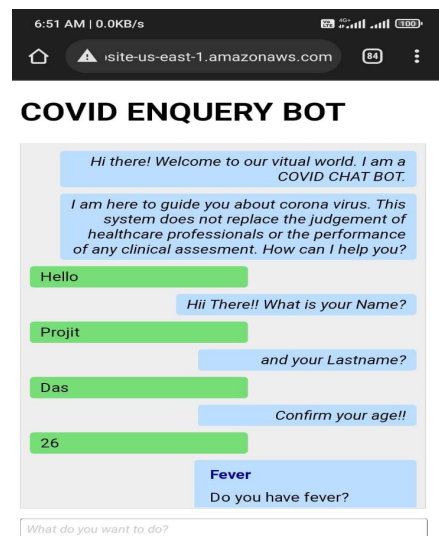


Figure 7. Response for COVID validation

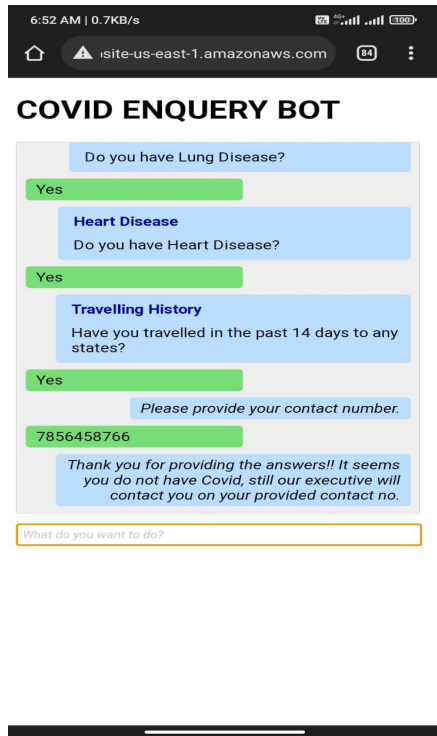


Figure 8. Response for COVID confirmation feedback

Here we have one option if a user wants to know about the information regarding COVID-19, what are the symptoms of COVID-19 and what precautions also? We can also get information regarding rumours and fake news. So, from this chatbot, users can also get information regarding that. Snapshots of this feature are given below; following Figures 9 and 10.

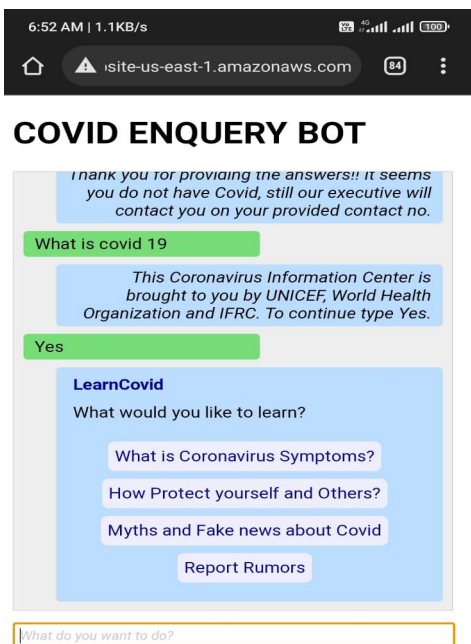


Figure 9. Response for COVID information 1

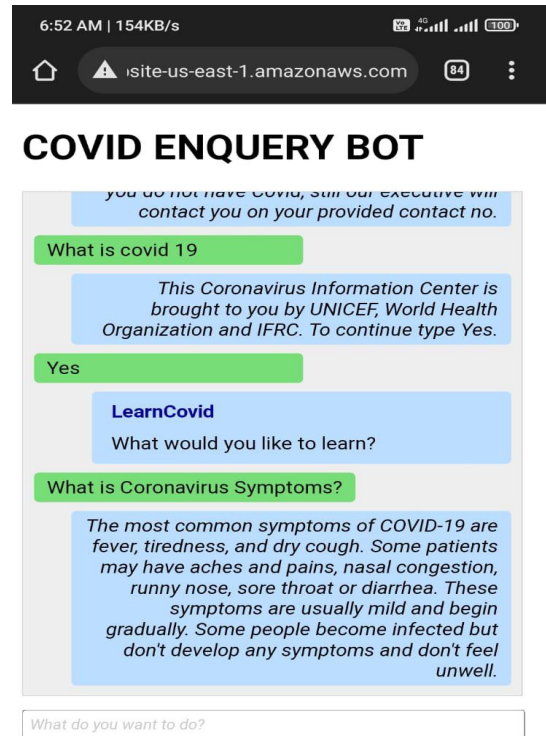


Figure 10. Response for COVID information 2

From this application, we also got some information related to the COVID situation in a different city. What are the COVID rate, and the active situation of a city we can easily get from this chatbot? Figures 11 and 12 describe this part.

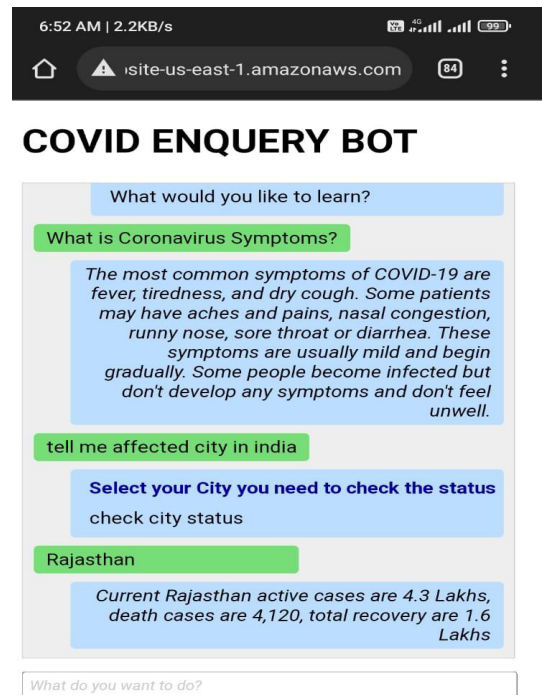


Figure 11. Choose a city to know COVID update

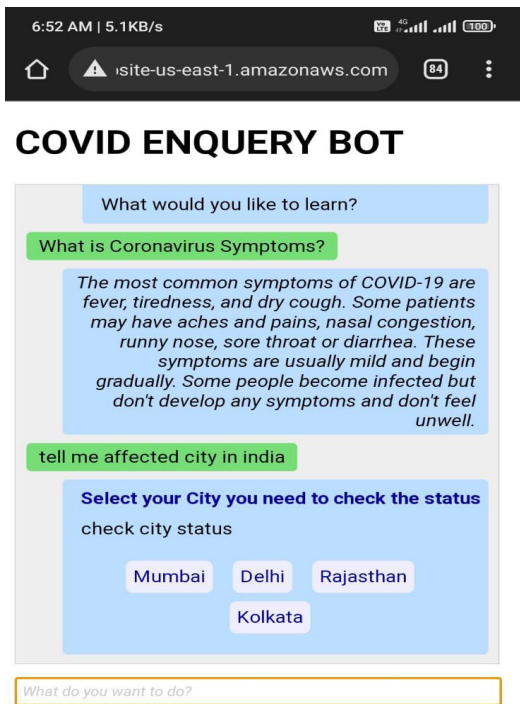


Figure 12. Response from chatbot

5. Conclusions

The design and execution of a clever chatbot application utilized to certify help and e-awareness during the COVID-19 epidemic are described in depth in this study. The purpose of this study is to develop smart chatbots that can provide consumers with information about the coronavirus during pandemics like COVID-19. A chatbot's goal is to start a real-world discussion. In our work, we suggest using the Amazon Lex, Cognito, and S3 approaches provided by Amazon Web Services in order to properly address customer inquiries. Our chatbot, CO-BOT, operates by going through three steps in succession: connecting, understanding, and replying. The user delivers a query via the user interface in plain text during the connection stage. Processing the user's question is enabled by the comprehension phase. Building the chatbot responses is the responsibility of the response stage. An English version is one of the two CO-BOT versions that are used and tested. In addition, CO-BOT can raise awareness of these disorders and offer support. This study takes an innovative approach to raise awareness of COVID-19 symptoms, preventing misunderstanding, taking precautions, providing facts and myths, reducing infection-restricting behaviours, and even assisting in mitigating the pandemic mental health burden. The key data have been compiled with the help of interviews conducted to promote this study. After reviewing the collected data, the researcher will assume

that the chatbot has been rated very highly on the basis of user experience. In terms of usability, it was found functional and simple by all the respondents. The chatbot is very accurate, well-formatted, and readily accessible due to the nature of the information delivered.

6. Future Scope

With the aid of the input provided from the interviews, the potential direction of the research study can be concluded. It can be inferred after data is evaluated that additional functions can be incorporated into the system. In future work, we want to compare our application's analytical performance against that of other COVID-19 applications and chatbots. We also intend to improve our chatbot by taking into account some extended versions viz.: 1) Include a module related to speech recognition; 2) Offer a virtual doctor meeting and best practices for a healthy lifestyle; and 3) Improve COVIBOT's data interpretation by incorporating features like graphical and pictorial representation. 4) Link a real-time database to obtain the most recent COVID status.

Conflict of Interests

The authors declare that they have no conflict of interests.

References

- [1] Gates, B., 2020. Responding to Covid-19-A once-in-a-century pandemic? *The New England Journal of Medicine*. 382(18), 1677-1679.
- [2] Sohrabi, C., Alsa, Z., O'Neill, N., et al., 2020. World health organization declares global emergency: A review of the 2019 novel coronavirus (COVID-19). *International Journal of Surgery*. 76, 71-76.
- [3] Thorp, H.H., 2020. A dangerous rush for vaccines. *Science*. 369(6506), 885.
- [4] Nehme, M., Braillard, O., Alcoba, G., et al., 2020. COVID-19 symptoms: Longitudinal evolution and persistence in outpatient settings. *Annals of Internal Medicine*. 174(5), 723-725.
- [5] Driss, M., Almomani, I., Alahmadi, L., et al., 2022. COVIBOT: A Smart Chatbot for Assistance and E-Awareness during COVID-19 Pandemic. *SMART-TECH 2022: The Second International Conference of Smart Systems and Emerging Technologies*; Apr 2022; Saudi Arabia. DOI: <https://doi.org/10.48550/arXiv.2204.07851>
- [6] Bharti, U., Bajaj, D., Batra, H., et al., 2020. Medbot: Conversational artificial intelligence-powered chatbot for delivering telehealth after covid-19. *5th Interna-*

- tional Conference on Communication and Electronics Systems (ICCES); 2020 Jun 10-12 ; Coimbatore, India.USA: IEEE. pp. 870-875. DOI: <https://doi.org/10.1109/ICCES48766.2020.9137944>
- [7] Ferreira, T.C., Marcolino, M.S., Ramos, I., et al., 2020. ANA: A Brazilian Chatbot Assistant about COVID-19. Proceedings of Conference ACL 2020 Workshop on Natural Language Processing for COVID-19 (NLP-COVID); 2020 Jul; Seattle, Washington, United States.
- [8] Dennis, A.R., Kim, A., Rahimi, M., et al., 2020. User reactions to COVID-19 screening chatbots from reputable providers. *Journal of the American Medical Informatics Association*. 27(11), 1727-1731.
- [9] Tiwari, V., Verma, L.K., Sharma, P., et al., 2021. Neural network and NLP based chatbot for answering COVID-19 queries. *International Journal of Intelligent Engineering Informatics*. 9(2), 161-175.
- [10] Abd-Alrazaq, A.A., Rababeh, A., Alajlani, M., et al., 2020. Effectiveness and safety of using chatbots to improve mental health: systematic review and meta-analysis. *Journal of Medical Internet Research*. 22(7), e6021.
- [11] Abd-Alrazaq, A.A., Alajlani, M., Ali, N., et al., 2020. Perceptions and opinions of patients about mental health chatbots: Scoping review. *Journal of Medical Internet Research*. 23(1), e17828.
- [12] Li, B., Wisniewski, G., Crabbé, B., 2021. Are Transformers a Modern Version of ELIZA? Observations on French Object Verb Agreement. Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing; 2021 Nov; Punta Cana, Dominican Republic. USA: Association for Computational Linguistics. p.4599-4610. DOI: <https://doi.org/10.18653/v1/2021.emnlp-main.377>
- [13] Colby, K.M., 1981. Modeling a paranoid mind. *Behavioral and Brain Sciences*. 4(4), 515-534.
- [14] Raj, A.B., Johnsen, K., Dickerson, R.F., et al., 2007. Comparing interpersonal interactions with a virtual human to those with a real human. *IEEE Transactions on Visualization and Computer Graphics*. 13(3), 443-457.
- [15] Fadhil, A., 2018. Beyond patient monitoring: Conversational agents role in telemedicine & healthcare support for home-living elderly individuals [Internet] [Submitted on 2018 Mar 3]. Available from: <https://arxiv.org/abs/1803.06000>
- [16] Amato, F., Marrone, S., Moscato, V., et al., 2017. Chatbots Meet eHealth: Automatizing Healthcare. *WIAIAH@ AI* IA. Computer Science*. 40-49.
- [17] Comendador, B.E.V., Francisco, B.M.B., Medenilla, J.S., et al., 2015. Pharmabot: A pediatric generic medicine consultant chatbot. *Journal of Automation and Control Engineering*. 3(2).
- [18] Josephine, D.D., 2021. Amazon Lex based personalized chatbot for IoT LAB@CIT. *International Journal of Advance Scientific Research and Engineering Trends*. 6(5).
- [19] Armstrong, S., 2018. The apps attempting to transfer NHS 111 online. *Technology in Healthcare*. DOI: <https://doi.org/10.1136/bmj.k156>
- [20] Holmes, S., Moorhead, A., Bond, R., et al., 2019. Usability testing of a healthcare chatbot: Can we use conventional methods to assess conversational user interfaces? Proceedings of the 31st European Conference on Cognitive Ergonomics; 2019 Sep 10. p. 207-214. DOI: <https://doi.org/10.1145/3335082.3335094>
- [21] Gupta, J., Singh, V., Kumar, I., 2021. Florence-a health care chatbot. 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS) ;2021 Mar 19-20; Coimbatore, India. USA: IEEE. p. 504-509.
- [22] Rogers, R., 2021. Internet of things-based smart healthcare systems, wireless connected devices, and body sensor networks in covid-19 remote patient monitoring. *American Journal of Medical Research*. 8(1),71-81.
- [23] Waheed, N., Ikram, M., Hashmi, S.S., et al., 2022. An Empirical Assessment of Security and Privacy Risks of Web based-Chatbots [Internet] [Submitted on 2020 May 17]. Available from: <https://arxiv.org/abs/2205.08252>
- [24] Bornholt, J., Joshi, R., Astrauskas, V., et al., 2021. Using lightweight formal methods to validate a key-value storage node in Amazon S3. Proceedings of the ACM SIGOPS 28th Symposium on Operating Systems Principles; 2021 Oct. pp. 836-850. DOI: <https://doi.org/10.1145/3477132.3483540>