# DEVELOPMENT OF A FUZZY INFERNENCE SYSTEM BASED WEB APPLICATION FOR EARLY PREDICTION OF CORONAVIRUS DISEASE

**ANUSHA KABIR**

**M. ENGINEERING THESIS**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**
**MILITARY INSTITUTE OF SCIENCE AND TECHNOLOGY**
**DHAKA, BANGLADESH**

**MARCH 2024**

# DEVELOPMENT OF A FUZZY INFERENCE SYSTEM BASED WEB APPLICATION FOR EARLY PREDICTION OF CORONAVIRUS DISEASE

ANUSHA KABIR (SN.0418140020)

A Project Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Engineering in Computer Science and Engineering

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
MILITARY INSTITUTE OF SCIENCE AND TECHNOLOGY
DHAKA, BANGLADESH

MARCH 2024

DEVELOPMENT OF A FUZZY INFERENCE SYSTEM BASED WEB APPLICATION FOR EARLY PREDICTION OF CORONAVIRUS DISEASE

M. Engineering Thesis

By

ANUSHA KABIR (0418140020)

Approved as to style and content by the Board of Examination on 25 March, 2024:

Assistant Professor Dr. Akhtaruzzaman — Chairman (Supervisor)
Department of Computer Science and Engineering
Military Institute of Science and Technology, Dhaka

Lt Col Muhammad Nazrul Islam, PhD — Member (Internal)
Class A Instructor
Department of Computer Science and Engineering
Military Institute of Science and Technology, Dhaka

Assistant Professor Dr. Md Towfiqur Rahman — Member (External)
Department of Computer Science and Engineering
University of Asia Pacific, Dhaka

Department of Computer science Engineering, MIST, Dhaka

DEVELOPMENT OF A FUZZY INFERENCE SYSTEM BASED WEB APPLICATION FOR EARLY PREDICTION OF CORONAVIRUS DISEASE

## DECLARATION

I hereby declare that the study reported in this project entitled above is my own original work and has not been submitted anywhere for any degree or other purposes. Further, I certify that the intellectual content of this project is the product of my own work and that all the assistance received in preparing this project and sources have been acknowledged and/or cited in the reference section.

Anusha Kabir
Student No: 0418140020

Department of Computer Science and Engineering, MIST, Dhaka

ACKNOWLEDGEMENTS

First and foremost, I would like to praise Allah the Almighty, the Most Gracious, and the Most Merciful for His blessing given to me during my study and in completing this thesis. May Allah's blessing goes to His final Prophet Muhammad (peace be up on him), his family and his companions.

I would like to express my heartfelt gratitude to the following individuals and organizations who have contributed to the completion of this research project:

I would like to express my deep and sincere thanks to my supervisor, Assistant Professor Dr. M. Akhtaruzzaman, Department of Computer Science and Engineering, Military Institute of Science and Technology (MIST), for providing invaluable guidance throughout this project. He has inspired me with his vision, integrity, and drive. It was a real honor and privilege to work under his direction.

I would like to thank the Department of Computer Science and Engineering (CSE) of the Military Institute of Science and Technology (MIST), Mirpur Cantonment, Dhaka, Bangladesh for supporting this project. A special thanks to Dr. Md. Mahbubur Rahman and Lt Col Dr. Muhammad Nazrul Islam for their valuable advice, criticism, and overall support.

Special gratitude goes to the Quantum Robotics and Automation Research Group (QRARG) and R&D section of DAZDREAM Robotics for providing technical support and insightful suggestions.

I want to express my sincere gratitude to my parents and family for their unwavering support and patience during my project.

Last but not least, I want to express my gratitude to everyone who helped me, directly or indirectly, to finish this project.

# ABSTRACT

**Development of a Fuzzy Inference System Based Web Application for Early Prediction of Coronavirus Disease**

COVID-19 or Coronavirus is the most crucial health crisis of present times globally. Starting from China every part of the world has fallen a victim to this pattern changing, life taking virus. Due to its high transmission power everyday thousands are attacked and dying, eventually leading to severe disruption in social, economic, academic arenas across the globe. All countries are in an unseen race to decline the rate of positive cases, spread and deaths caused by disease through isolation and treatment. The most prominent way of achieving this is testing. But people are mostly unaware or in dilemma about testing as maximum symptoms of COVID-19 coincides with regular flue. Delay in identifying COVID-19 may result in postponed medical intervention, enabling the virus to advance unchecked and potentially induce severe respiratory complications. Prompt diagnosis plays a pivotal role in commencing timely treatment, averting complications, and mitigating the risk of virus-related mortality. Consulting doctors or going for tests puts them in vulnerable situation and exposes them to the disease, this being a highly infectious one. While staying untested is risky too. To help in this scenario this study proposes a fuzzy inference system-based model to predict the possibility of being COVID-19 positive based on symptoms provided as input by users to the model. Fuzzy being a robust model, with capability to understand and process vague and imprecise data, is better suited over other models to interpret human inputs of COVID-19 symptoms. Different membership functions have been used to map the symptoms to fuzzy values. Then various defuzzification methods are used to observe the result and to perform a comparative study. From the comparative analysis, COA method has proven the best fitted result. Thus, COA has been used for the web application tool. The system outputs the possibility in the considered scale and suggests the need of test. The result of this system is interpreted by identifying cases in three separate categories: Not Needed, Stay Isolated and Test Immediately. An output score between 0 to 3 indicates that test is not needed. Output score between 2 to 5 provides indication to stay isolated and beyond 5 is a state where test is recommended to be done immediately.

ABSTRACT

**Development of a Fuzzy Inference System Based Web Application for Early Prediction of Coronavirus Disease**


কোভিড-১৯ বা করোনাভাইরাস বিশ্বব্যাপী আমাদের সময়ের সর্বাধিক গুরুত্বপূর্ণ স্বাস্থ্য সঙ্কট। চীন থেকে শুরু করে এই ভাইরাসের প্রভাব প্রত্যক্ষ বা অপ্রত্যক্ষভাবে সমস্ত দেশে প্রকাশ পেয়েছে। এর উচ্চ সংক্রমণ শক্তি দিয়ে প্রতিদিন হাজার আক্রান্ত হচ্ছে এবং মারা যাচ্ছে। সব দেশ পরিস্থিতি সামাল দেয়ার জন্য, এ অবস্থা থেকে পরিত্রাণ পাওয়ার জন্য, সংক্রমণ এবং মৃত্যু হার কমানোর জন্য, রোগ পরীক্ষা এবং চিকিৎসাকে অত্যাধিক গুরুতবের সাথে দেখছে। তবে, অনেকে নিয়মিত জ্বরের সাথে কভিড-১৯ এর লক্ষণগুলো ভিন্ন করতে না পেরে অনিশ্চিত অবস্থায় থাকে। কোভিড-১৯ শনাক্তে বিলম্ব ঘটলে চিকিৎসা শুরুতে দেরি হতে পারে, ফলে ভাইরাসটি আরও ব্যাপকভাবে ছড়ানোর সুযোগ পায় এবং শ্বাসযন্ত্রে গুরুতর জটিলতা তৈরি করতে পারে। সময়মতো চিকিৎসা শুরু করতে, জটিলতা এড়াতে এবং ভাইরাসজনিত মৃত্যুর ঝুঁকি কমাতে দ্রুত নির্ণয় অত্যন্ত গুরুত্বপূর্ণ। তবে, ডাক্তারের সাথে পরামর্শ কিংবা পরীক্ষা করাতে গেলে সংক্রমণের ঝুঁকি থাকে, কিন্তু পরীক্ষা না করিয়ে থাকলেও ঝুঁকি রয়েছে। এই গবেষণায় কোভিড -১৯ আক্রান্ত হওয়ার সম্ভাবনা পূর্বাভাস করার জন্য একটি ফাজি ইনফারেন্স সিস্টেম-ভিত্তিক মডেল প্রস্তাব করা হয়েছে। ব্যবহারকারীরা মডেলে যেসব উপসর্গের তথ্য দেন, সেগুলোর উপর ভিত্তি করে এই সম্ভাবনা নির্ণয় করা হয়। ফাজি একটি শক্তিশালী মডেল, যা অস্পষ্ট তথ্য বোঝার এবং প্রসেস করার ক্ষমতা রাখে। মানুষের কোভিড-১৯ লক্ষণগুলির তথ্য বিশ্লেষণে অন্যান্য মডেলের তুলনায় ফাজি তার সক্ষমতার প্রমাণ দিয়েছে । বিভিন্ন সদস্যতা ফাংশন ব্যবহার করা হয়েছে লক্ষণগুলির ফাজি মান দেখার জন্য এবং তত্ত্বপ্রযুক্তির পরিপ্রেক্ষিতে COA পদ্ধতি নির্বাচন করা হয়েছে। শেষ পরিণামে সিস্টেম বিবেচনা করে পরীক্ষা প্রয়োজনীয়তা সুপারিশ করে। এই সিস্টেমের ফলাফল তিনটি পৃথক বিভাগে বিভক্ত করে বোঝানো হয়েছে: প্রয়োজন নেই, অবস্থান পৃথক করুন, এবং তাড়াতাড়ি পরীক্ষা করুন । ০ থেকে ৩ এর মধ্যে একটি আউটপুট স্কোর দেখায় যে পরীক্ষা প্রয়োজন নেই। ২ থেকে ৫ এর মধ্যে আউটপুট স্কোর আলাদা থাকতে ইঙ্গিত দেয় এবং ৫ এর অতিরিক্ত আউটপুট স্কোর একটি অবস্থা যেখানে পরীক্ষার পরামর্শ দেওয়া হয়।

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

# CHAPTER 1
# INTRODUCTION

Corona is an infectious disease caused by the most recently discovered virus called COVID-19. After its first outbreak in Wuhan, China in 2019 (Zhu et al., 2020), in no time it has been identified as pandemic by World Health Organization, WHO. Millions of cases and fatalities have been reported because of the COVID-19 pandemic, which has devastated the whole world. Though COVID-19 can affect all, being a heavily contagious disease, but for people in elderly age group and with pre-existing health issues, COVID-19 is more likely to develop severe condition and even may lead to death. According to WHO (Bangladesh: WHO Coronavirus Disease (COVID-19) Dashboard with Vaccination Data, n.d.), till now the death count reported globally is over 6.9M, and in Bangladesh this crossed 29.4K.

## 1.1. Background of the project

COVID-19 virus's unpredictable nature and the introduction of new varieties have made it difficult to forecast the course of the pandemic, despite continued efforts to prevent its spread. The pandemic has emerged as a global health crisis, necessitating effective strategies for prediction and containment to mitigate its impact on public health. The most common symptoms of COVID-19 are fever, dry cough, loss of taste and smell, fatigue, headache etc. But most people are unaware of existence of COVID-19 in body with presence of mild symptoms. Many more failing to differentiate the symptoms of COVID-19 and regular flue, remains in a state of indecision regarding whether to go for test or not. Some avoid going for test assuming exposure to the disease and some due to ignorance. As a result, COVID-19 grows in body, destroys lungs, and remains undetected until the very last stage, late detection of COVID-19 can lead to delayed medical intervention, allowing the virus to progress unchecked and potentially causing severe respiratory complications which is eventually contributing to the rise of the death rate.

Besides, since the beginning of the pandemic, several prominent variants have been seen, like Alpha, Beta, Delta, and Omicron. Understanding and monitoring symptoms of each new variant that surfaces is essential in ensuring preparedness, as new variants are an expected part of virus evolution. Alpha was the first highly publicized variant globally, believed to be 30 to 50% more contagious than the original SARS-CoV-2 strain. Beta was about 50% more contagious than the original strain, and there was evidence suggesting it may have been more likely to lead to hospitalization and death than other variants. Omicron's subvariants are considered especially efficient spreaders of the disease. The original Omicron strain was more transmissible than Delta. Delta caused more severe disease in unvaccinated individuals and was more likely to result in hospitalization.

During the pandemic, COVID-19 infection rates had been changing because of mass vaccination, organic infections, and the evolution of new variants. While in the early stages, infections mostly affected fully susceptible individuals, breakthrough infections among vaccinated people and reinfections among those previously infected were also observed later. In many cases lesser severity of symptoms had been reported for vaccinated individuals. However, vaccinated individual can still have viral load (amount of virus produced by an infected person), though lower from that of non-vaccinated people and contribute to spreading the disease. So basic awareness and understanding about relevant symptoms is an important part of preparedness to combat the disease.

## 1.2. Problem Statement

To address the multifaceted challenges posed by COVID-19, it is important to acknowledge and comprehend the various clear complications and impacts of the virus. From the emergence of new strains such as Delta and Omicron to the varied representation of symptoms and transmission dynamics, the landscape of COVID-19 is constantly evolving. Considering these complexities,

- there is a dire need to provide remote comprehendible assistance through eHealth services to aid individuals in identifying potential health threats,
- the accessibility and reliability of such services remain inconsistent, often resulting in prolonged wait times for scheduled appointments with healthcare providers,
- the inherent imprecision and vagueness of input data puts forward significant challenges in accurately predicting the course of the pandemic.

Therefore, establishing a platform that facilitates the seamless sharing of symptoms and delivers timely feedback could greatly help mitigating these challenges, providing individuals with convenient access to medical guidance and support. Furthermore, prevailing forecasting methodologies encounter persistent hurdles starting from data variability, model accuracy limitations, and adaptability issues. To overcome these obstacles, an innovative approach is needed—one that can accommodate linguistic variables and incorporate human-like reasoning mechanisms with flexibility. By enhancing the intuitiveness and interpretability of the forecasting process, such an approach can empower decision-makers to grab the complexities of COVID-19 more effectively. In this dynamic and uncertain landscape, human interpretation plays a pivotal role in guiding decision-making, making it essential to develop forecasting models that are both accurate and easily comprehensible.

### 1.3. Significance of project and applied method

The Fuzzy Inference System (FIS) presents a sophisticated framework for integrating linguistic variables and fuzzy sets, thereby enhancing the modeling of ambiguous and uncertain medical scenarios with greater efficacy. The term "fuzzy" denotes the system's capability to effectively manage imprecise or indeterminate input data.

Basically, a fuzzy logic controller creates a language-based system or a linguistic framework that connects what it sees as input with what it does as output, skipping the complicated math usually involved (Jaekel et al., 2004). It's worth noting that the various types of Fuzzy controllers diverge in their approach towards implementing the inference engine and the methodology employed for defuzzification.

In contrast to alternative uncertainty-handling mechanisms, Fuzzy controllers boast numerous advantages, notably their relatively low demand for computational resources during implementation. A FIS is indeed a type of artificial intelligence strategy that can be data-efficient compared to some other machine learning techniques. Unlike many traditional machine learning algorithms that require large amounts of labeled data for training, fuzzy logic-based systems can operate effectively with just a few rules and a limited dataset. By integrating a Fuzzy inference system into the backend of a web application, a robust platform can be developed capable of effectively managing, interpreting, and translating human input while delivering precise predictions. Such a system represents a timely solution, particularly in the context of probable COVID-19 cases, facilitating decision-making processes concerning subsequent steps in the treatment protocol.

Furthermore, the utilization of FIS within a web application framework not only streamlines the handling of uncertain medical data but also holds the potential to significantly impact patient care. Through its ability to swiftly analyze and interpret complex information, the system empowers healthcare professionals to make informed decisions in real-time, thereby enhancing the overall efficiency and efficacy of the treatment process for individuals potentially affected by COVID-19.

### 1.4. Objectives

The objectives of the project are:

- To develop a fuzzy inference system for early prediction of Coronavirus disease.
- To analyze the results and validate through several test case scenarios.
- To develop a web application with the designed fuzzy based system.

### 1.5. Methodology

The methodology can be summed up in the following steps:

- Conducting extensive literature review to identify existing disease prediction systems with similar mechanism.
- Determining input and output for the system
- The inputs will be fuzzified using membership functions.
- Determining the fuzzy rule set.
- Rules aggregation will be performed through various defuzzification techniques to obtain crisp output.
- Design of output membership function

- Validating the output, revisiting rule set if result is not satisfactory.
- Implementing this FIS model as backend of Web GUI.
- Testing the web tool to verify and validate the results.

Fig 1.1 shows the methodology of the project.

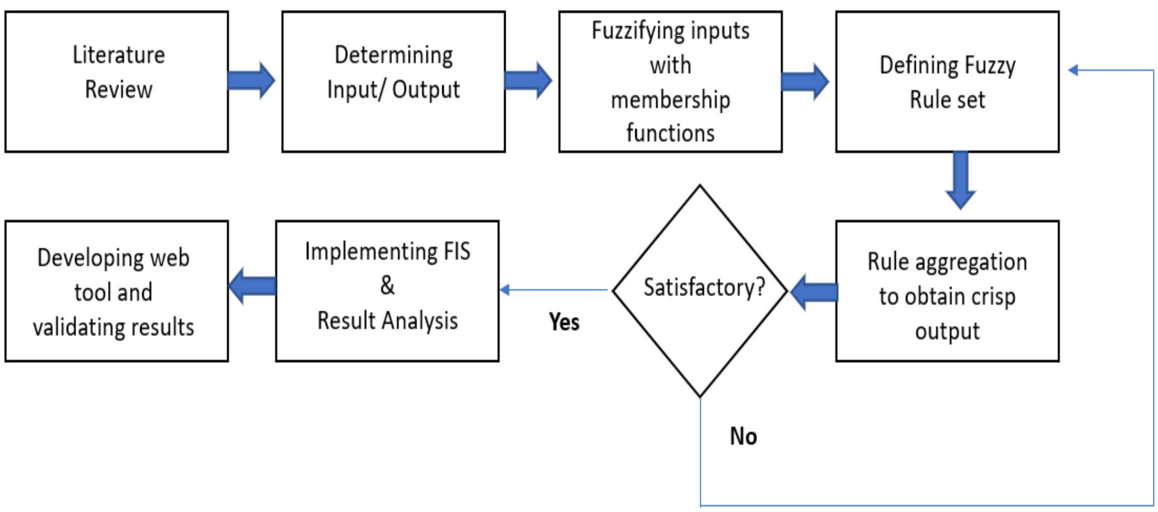


Figure 1.1: Methodology of project

## 1.6. Organization of the Remaining Chapters

The rest is organized in the following order:

In Chapter 2, the literature review is presented where several disease prediction mechanism and application of Fuzzy inference system is discussed.

In Chapter 3, a detail design of proposed system is given along with membership function and defuzzification techniques applied.

In Chapter 4, System setup and implementation are discussed.

In Chapter 5, Result analysis, discussion and comparative analysis are presented.

In Chapter 6, the project is concluded, and scope of future works are discussed.

# CHAPTER 2
# LITERATURE SURVEY

In recent years, the global landscape has been profoundly shaped by the unprecedented challenges posed by the COVID-19 pandemic. The urgency to comprehend, mitigate, and combat the far-reaching impacts of this novel coronavirus has fueled an array of diverse and comprehensive research efforts worldwide. This chapter carefully looks at key studies about COVID-19 and Fuzzy inference system, each adding something unique to what is known and helping to find what research still needed to be done.

## 2.1. Previous Work

With a view to understanding better, the mechanism of fuzzy system and its application in the sector of disease risk prediction, some studies have been performed. Infectious diseases are no less than a severe crisis for human life, in these scenarios there is no alternative to proactive diagnosis. The need of appropriate and dependable prediction of infectious diseases is focused on this study (Erraguntla et al., 2017b). Authors also proposed an integrated framework for infectious disease analysis that counts the whole disease life cycle including pre-emergence, emergence and spreading state. In another research (Vinarti & Hederman, 2019) authors have put forward a prediction system that predicts the risk of being infected based on environmental and individual human factors. The system has knowledge representation that holds the epidemiological knowledge of infectious diseases in an encoded form and an algorithm that makes a model from this knowledge for predicting the risk of diseases in terms of probability.

Based on the need of predicting possibility or probability of being infected by a disease, different works have been done. A summary of different fuzzy systems for disease prediction that have been studied is presented here. Authors have implemented fuzzy rule base in combination with decision tree for the prediction of heart disease (Pathak & Valan, 2019). Eight attributes collected from patients have been used as input for the system and an accuracy of 88% have been achieved. A clinical decision support system based on weighted fuzzy concept is discussed in (Sharma & Saxena, 2017) for coronary illness. The system uses both fuzzy logic and genetic algorithm and is composed of two stages. First stage has a mechanical methodology of weighted fuzzy rules and second stage consists of model combining the two approaches. Omregbe et al. in their work (Omoregbe et al., 2020) presented a text messaging-based system for assessing the several symptoms of tropical diseases present in Nigeria. They have used Telegram Bot API for connecting human and the system and on the contrary Twillo was used for connecting system and human. Finally, a fuzzy SVM was used for the backend model.

A Clinical Decision Support System (CDSS) has been proposed in a study by Tandra et al. (Tandra et al., 2020). It performs decision making for common diseases like malaria, typhoid, dengue and other epidemic prone diseases. Based on the severity of the symptoms an adaptive neuro fuzzy inference system (ANFIS) performs the diagnostic decision making. The design of a hybrid heart disease detection system has been explained in the research work by Gadekallu and Khare (Gadekallu & Khare, 2016). It uses OFBAT and rule based-Fuzzy logic (RBFL). It used locality preserving projection (LPP) for selecting the most relevant features from chosen data set. Then rules of fuzzy logic were created from this data. A hybrid neuro fuzzy system (HNFS) design is presented in the study by Thiyagarajan (Thiyagarajan, 2017) that predicts different lung disease like asthma, lung cancer or tuberculosis. For the purpose of data collection, Cancer Assessment Questionnaire was prepared and provided, that aided in symptom obtaining. Using Pearson’s correlation, the most vital symptoms were picked. The final output was compared with gold standard results and were seen to be satisfactory. A fuzzy expert system for heart disease prediction is shown in the work by Sharma and others (Sharma et al., 2018b). Taking 19 parameters as input, the system outputs the chances of having heart disease in terms of low, medium and high. For the development of this system, 3 different rule bases are made based on symptoms and blood test result. Kaur and others proposed a genetic neuro fuzzy system that diagnoses hypertension (Kaur et al., 2014). Considering inputs like BMI, cholesterol, blood urea etc., the output is given as low, medium, high risks. The result of the proposed system in comparison to neuro fuzzy system is seen to perform better. Combining the concepts of neural network and fuzzy system, a hybrid AI model has been used to which diagnoses breast cancer with the help of defined fuzzy rule base (Araújo et al., 2019). The feasibility was tested using binary pattern classification and the result obtained showed that the system performed with good level of accuracy in prediction. Another research work

(Kanimozhi et al., 2018) also predicts breast cancer using a fuzzy temporal rule based intelligent model. This fuzzy temporal rules also helped in determining the most significant features from the dataset. The system has been evaluated with the help of domain experts and is mentioned to be better than existing system for diagnosis.

Also, at present times, fuzzy system and concept have been used for the different types of prediction related to COVID-19. An intelligent approach for predicting COVID-19 time series is explained in a work by Castillo and Melin (Castillo & Melín, 2020). It is a hybrid approach that combined fractal theory and fuzzy logic. Fuzzy logic has been used in the system to put forward the presence of uncertainty in the forecast processing. Linear and nonlinear fractal time series dimensions are the inputs to the fuzzy system and provides as output country wise forecast from the confirmed death and positive cases of COVID-19 time series. Another research (Cihan, 2020) shows a fuzzy rules-based system that predicts daily COVID-19 cases. They implemented Wang and Mendel's fuzzy rule method to find the number of validated cases in the outbreak of this disease. Another interesting work (Asl et al., 2021) predicts the ICU admission among COVID-19 infected patients. To deal with the uncertainty of proper decision making regarding optimal resource allocation, fuzzy logic has been used. The authors developed an interval type-2 FES and also an adaptive neuro fuzzy inference system ANFIS. Authors came up with a IoT based technique enabled with FIS for monitoring and predicting if a human has been infected (Fatima et al., 2020). In this system, FIS used to in the prediction layer for output prediction. If the inputs taken from the sensors are seen to be correct, it will be fuzzified and flow through the FIS for output generation. Design of a system that predicts existence of COVID-19 in patient is presented in a study (Khamparia et al., 2021). It implements ANFIS which provides the best of ANN. Another study (Onari et al., 2021), proposed a self-assessment decision support system that helps to differentiate the severity level COVID-19 positive patients to provide best care. Data driven Bayesian Network and Fuzzy Cognitive Map is merged to form the DSS. Here data is categorized in three level of severity. Then Bayesian network segregates the EBP relationships present among the symptoms. Then the ranking of these symptoms is done by FCM. In another exploration, using fuzzy tsukamoto inference system, prediction of recovery risk rate of COVID-19 is explained (Wulandari et al., 2021). Using age and comorbid as input it gives output of mortality rate by centroid triangular fuzzy number.

From the above discussion, some notable research works on disease prediction using fuzzy inference system is summarized below in Table 2.1 which will direct us to better understand the research gap that has been addressed through the work

Table 2.1: Existing Research Works and their Limitation(s)

| Research Work | Approach | Limitation |
|---|---|---|
| **Pathak & Valan, 2019** | Fuzzy rule base combined with decision tree for heart disease prediction | No medium developed for mass people to access the outcome |
| **Gadekallu & Khare, 2016** | Hybrid heart disease detection system using OFBAT and rule-based fuzzy logic (RBFL) | Developed specifically for doctors to automate heart disease diagnosis and not accessible for mass |
| **Thiyagarajan, 2017** | Hybrid neuro fuzzy system (HNFS) for predicting lung diseases | Designed for physicians to identify lungs cancer at an early stage |
| **Castillo & Melín, 2020** | Intelligent approach combining fractal theory and fuzzy logic for predicting COVID-19 time series | Focusing on predicting time series only |
| **Wulandari et al., 2021** | Prediction of recovery risk rate of COVID-19 using fuzzy Tsukamoto inference system | Focusing on predicting recovery risk rate |

Major findings of relevant research works have been mentioned in brief in table below.

Table 2.2: Major findings of relevant research works

| Study | Methodology | Main Findings |
|---|---|---|
| Erraguntla et al., 2017b | Integrated framework for infectious disease analysis | Highlighted the need for accurate prediction of infectious diseases throughout their life cycle, including pre-emergence, emergence, and spreading state. |

| | | |
|---|---|---|
| Vinarti & Hederman, 2019 | Prediction system based on environmental and individual human factors | Developed a prediction system incorporating environmental and individual human factors to assess the risk of disease infection. |
| Pathak & Valan, 2019 | Fuzzy rule base combined with decision tree for heart disease prediction | Achieved an accuracy of 88% in predicting heart disease using fuzzy rule base in combination with decision tree methodology. |
| Sharma & Saxena, 2017 | Weighted fuzzy concept for coronary illness prediction | Developed a clinical decision support system for coronary illness using weighted fuzzy concept and genetic algorithm, achieving improved diagnostic accuracy. |
| Omoregbe et al., 2020 | Text messaging-based system for assessing symptoms of tropical diseases | Implemented a text messaging-based system using Telegram Bot API and Twilio for assessing symptoms of tropical diseases, with backend model using fuzzy SVM. |
| Tandra et al., 2020 | Adaptive neuro fuzzy inference system (ANFIS) for common diseases diagnosis | Proposed a Clinical Decision Support System (CDSS) using ANFIS for diagnostic decision-making of common diseases such as malaria, typhoid, and dengue. |
| Gadekallu & Khare, 2016 | Hybrid heart disease detection system using OFBAT and RBFL | Developed a hybrid heart disease detection system using locality preserving projection (LPP) and rule-based fuzzy logic, achieving enhanced diagnostic accuracy. |
| Thiyagarajan, 2017 | Hybrid neuro fuzzy system for lung disease prediction | Designed a hybrid neuro fuzzy system for predicting lung diseases such as asthma, lung cancer, and tuberculosis, using a Cancer Assessment Questionnaire for data collection. |
| Sharma et al., 2018b | Fuzzy expert system for heart disease prediction | Developed a fuzzy expert system for heart disease prediction based on 19 input parameters, providing risk assessment in terms of low, medium, and high categories. |
| Kaur et al., 2014 | Genetic neuro fuzzy system for hypertension diagnosis | Proposed a genetic neuro fuzzy system for hypertension diagnosis, achieving improved performance compared to traditional neuro fuzzy systems. |

| | | |
|---|---|---|
| Araújo et al., 2019 | Hybrid AI model for breast cancer diagnosis | Developed a hybrid AI model combining neural network and fuzzy system for breast cancer diagnosis, demonstrating high accuracy in prediction. |
| Kanimozhi et al., 2018 | Fuzzy temporal rule-based intelligent model for breast cancer prediction | Designed a fuzzy temporal rule-based intelligent model for breast cancer prediction, incorporating significant features selection from the dataset. |
| Castillo & Melín, 2020 | Hybrid approach for COVID-19 time series prediction | Proposed a hybrid approach combining fractal theory and fuzzy logic for predicting COVID-19 time series, considering uncertainty in the forecast processing. |
| Cihan, 2020 | Fuzzy rules-based system for daily COVID-19 cases prediction | Developed a fuzzy rules-based system using Wang and Mendel's method for predicting daily COVID-19 cases, providing insights into disease outbreak trends. |
| Asl et al., 2021 | Interval type-2 FES and ANFIS for ICU admission prediction | Utilized fuzzy logic and ANFIS for predicting ICU admission among COVID-19 patients, addressing uncertainty in decision-making for resource allocation. |
| Fatima et al., 2020 | IoT-enabled FIS for COVID-19 monitoring and prediction | Developed an IoT-based technique with FIS for monitoring and predicting COVID-19 infection, leveraging sensor data for accurate prediction. |
| Khamparia et al., 2021 | ANFIS-based system for COVID-19 diagnosis | Implemented ANFIS for diagnosing COVID-19 infection, providing reliable diagnostic capabilities. |
| Onari et al., 2021 | Decision support system for severity level differentiation of COVID-19 patients | Proposed a self-assessment decision support system using Bayesian Network and Fuzzy Cognitive Map for differentiating severity levels of COVID-19 patients, facilitating optimized care delivery. |
| Wulandari et al., 2021 | Fuzzy Tsukamoto inference system for COVID-19 recovery risk rate prediction | Developed a fuzzy Tsukamoto inference system for predicting the recovery risk rate of COVID-19 patients based on age and comorbidities. |

## 2.2. Research Gap

From above studies, several implementations of disease prediction algorithms and approaches have been observed. Cases specific to COVID-19 prediction and application of fuzzy logic have also been explored. It is observed that very few works have been done on finding the possibility of being COVID-19 infected from prevailing symptoms in combination with Fuzzy logic. At the same time leveraging this COVID-19 prediction mechanism for ends users and making it easily available and accessible medium like application or web portal was not much focused. Thereby a scope to contribute to creating mass awareness among people regarding tests using an application was discovered. So, this study focuses on developing a fuzzy rule base system to output the possibility of being infected from the existing symptoms along with a web portal development to help people comprehend their severity level through a self-service modality.

## 2.3. Summary

Henceforth, within the confines of this chapter, a comprehensive and meticulous examination has been undertaken to scrutinize an array of studies delving into COVID-19 and, more specifically, Fuzzy logic. The primary aim of this endeavor is to derive insights into the current landscape, discern any prevailing lacunae, and endeavor to address them through the research at hand. At the heart of this study is the idea of making a web app that uses fuzzy logic to determine the possibility of being infected by coronavirus early. This app is expected to understand different ways people talk, that is human language and help figure out how bad COVID-19 cases are, so doctors can act fast and provide timely medical intervention.

# CHAPTER 3
# DESIGN OF THE SYSTEM ARCHITECTURE

Many health challenges have been solved successfully using artificial intelligence approaches. A precise model for biological systems might not exist or might be too challenging to construct. In such situations, fuzzy logic is considered to be ideal since control strategies are flexible enough to incorporate and convey human knowledge and experience.

Mamdani Fuzzy inference model has been chosen for the system implementation for its ease of converting output into linguistic form (Abdalla et al., 2018). Below the different components of the system along with their implementation detail have been mentioned

## 3.1. Fuzzy inference system (FIS) and membership function

FIS is a framework that simulated a system's behavior as IF-THEN rules using the expertise of experts or the system's historical data (Kazeminezhad et al., 2005). Conceptually Fuzzy inference system consists of:

- A rule base, which has a set of IF-THEN rules
- A database defining the membership functions
- A decision-making unit for performing the inference procedure
- A fuzzification interface for fuzzifying crisp input
- A defuzzification interface for de-fuzzifying inference results into crisp output

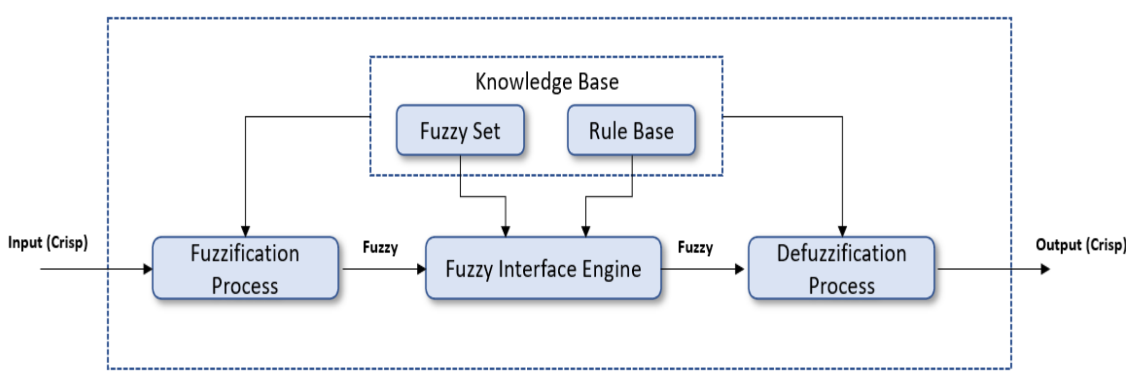


Figure 3.1: Fuzzy Inference System Architecture

Figure 3.1 depicts the architecture of the fuzzy inference system.

Membership function is a function specifying how an element x in the universe of discourse X has a membership value between 0 and 1 assigned to it. It is represented as follows:

$$A = \{x, \mu_A(x) | \, x \in X\} \quad \text{.........................................} (3.1)$$

Here fuzzy set A in the universe of discourse X is defined as a set of ordered pairs and $\mu_A(x)$ is the membership function of fuzzy set A.

There are different types of membership functions. Membership functions used in this project are discussed below:

**Triangular membership function:** The Triangular MF block implements a triangle-shaped membership function. Triangular membership function, μ_A (x) is defined by below equation 3.2:

$$\mu_A(x) = \begin{cases} 0 & x < \alpha_{min} \text{ or } x > \alpha_{max} \\ \frac{x-\alpha_{min}}{\beta-\alpha_{min}} & x \in (\alpha_{min}, \beta) \\ \frac{\alpha_{max}-x}{\alpha_{max}-\beta} & x \in (\beta, \alpha_{max}) \end{cases} \quad \text{..................................} (3.2)$$

Here, $\alpha_{min}, \alpha_{max}$ denotes left and right base points of the triangle, and the parameter β denotes the location of triangle peak as shown in in below figure.

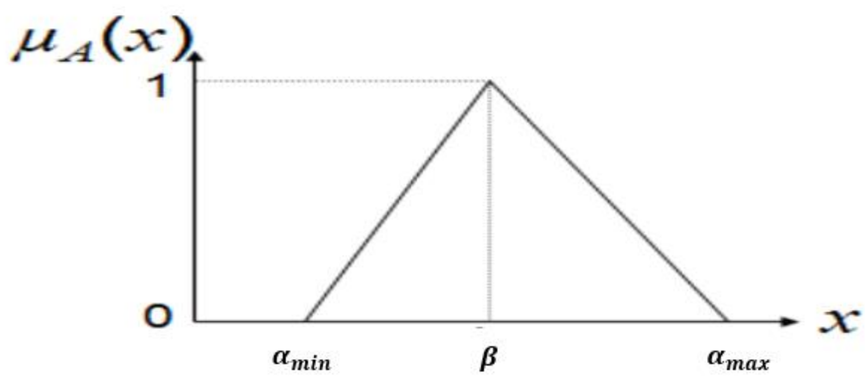


Figure 3.2: Triangular Membership Function

**Trapezoidal membership function:** The Trapezoidal MF block implements a trapezoid-shaped membership function. It is defined by below equation:

$$\begin{cases} 0 & x \leq \alpha_{min} \text{ or } x \geq \alpha_{max} \\ \frac{x-\alpha_{min}}{\beta_1-\alpha_{min}} & x \in (\alpha_{min}, \beta_1) \\ \frac{\alpha_{max}-x}{\alpha_{max}-\beta_2} & x \in (\beta_2, \alpha_{max}) \\ 1 & \beta_1 \leq x \leq \beta_2 \end{cases} \quad \ldots\ldots (3.3)$$

Here $\alpha_{min}$, $\alpha_{max}$ are parameters referring to base points of the trapezoid, while $\beta_1$, $\beta_2$ control the left and right shoulders or top points of the trapezoid as shown in below figure:

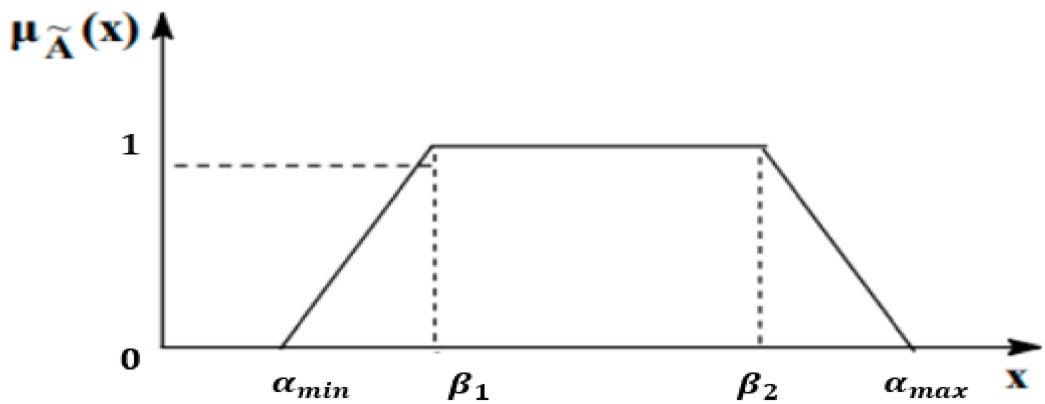


Figure 3.3: Trapezoidal Membership Function

**Singleton membership function:** An output function that is given by a spike at a single number rather than a continuous curve. It is defined by below equation:

$$\mu(x) = \begin{cases} 1, & x = c \\ 0, & Otherwise \end{cases} \quad \ldots\ldots (3.4)$$

It is represented by the impulse function as shown in figure below:

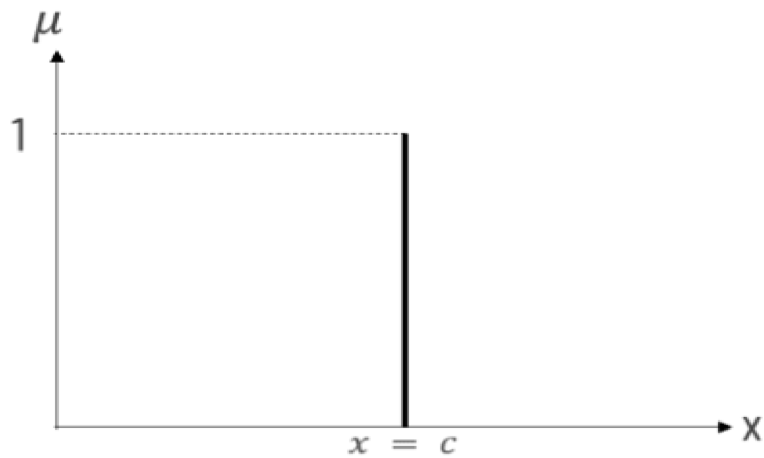


Figure 3.4: Singleton Membership Function

Both triangular and trapezoidal membership functions can be simply implemented and are computationally efficient. If there is no prior requirement of shapes, either of these can be chosen to implement membership functions.

For the purpose of developing the possibility generating fuzzy system Mamdani fuzzy inference system approach have been chosen. The FIS is implemented following below steps:

**Step 1:** Determining input and output for the system.

**Step 2:** Inputs are fuzzified using membership functions.

**Step 3:** Defining output membership function.

**Step 4:** Determining the fuzzy rule set.

**Step 5:** Rule aggregation

**Step 6:** Applying various defuzzification technique.

**Step 7:** Obtaining crisp output.

The proposed system architecture looks as follows in Figure 3.5:

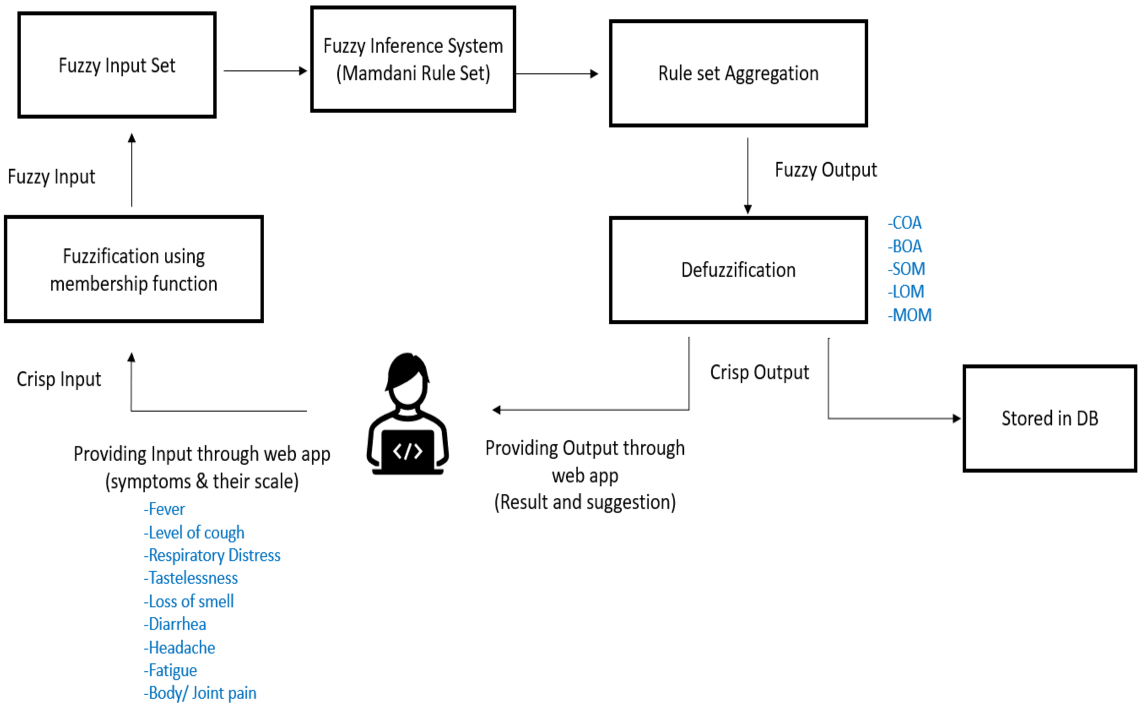


Figure 3.5: Proposed System Architecture

## 3.2. Determining input and output

A list of intensity of vital symptoms of COVID-19 is taken as input for the FIS. The symptoms chosen are:

- Fever
- Level of cough
- Respiratory distress
- Loss of smell
- Tastelessness
- Fatigue
- Headache
- Diarrhea
- Body/ Joint pain

Fever is given as input in Fahrenheit scale, ranging from 96°F to 104°F. Level of cough and Respiratory distress is given as input in a range of 0 to 5, where 0 indicates absence of the symptoms and 5 indicates most vividly present. Loss of smell, tastelessness, fatigue, headache, diarrhea and body or joint pain are all given as input in terms of 0 and 1 where 0 indicated absence and 1 indicates presence of the symptom.

The output of the system prior to applying defuzzification will be any of the below 3:

- Test not needed.
- Stay isolated and observe.
- Test immediately.

Post defuzzification number in a scale 0 to 10 will be obtained that will indicate the possibility of being covid infected. Where 0 indicates the least possibility and 10 indicates highest possibility.

### 3.3. Defining membership function for input

For each of the input variable, membership functions are defined as follows:

#### 3.3.1. Fever

Fever stands prominently as a hallmark symptom in the identification of COVID-19. Elevated body temperature, often accompanied by chills and sweating, serves as a crucial indicator of viral infection. The membership function for any symptom can be of any shape and form as long as it maps the given data with desirable degree of memberships. The choice of membership function depends on developers to decide. This is where fuzzy system offers individual degrees of freedom.

For fever triangular membership function has been chosen. Triangular membership function is one of the most encountered membership functions in practice and have the advantage of simplicity. Usually fever ranges from 100°F to 104°F. Figure 3.6 shows the membership function. 96°F to 99°F is considered as absent, 98°F to 102°F is considered as low and above

101°F is considered as high. Triangular shapes represent fuzzy numbers, while trapezoid shapes represent fuzzy intervals. However, here in this case trapezoidal membership function could also be chosen, but since it's used for some other symptoms, for variation here triangular membership function was selected.

### 3.3.2. Level of cough

Cough emerges as a distinctive identifying feature in the context of COVID-19, serving as a prevalent respiratory symptom associated with the viral infection. A persistent, dry cough is often observed in individuals infected with the coronavirus, distinguishing it from other respiratory illnesses. For level of cough, trapezoidal membership function has been used, to depict fuzzy interval clearly. Because trapezoid shape helps to better understand ranges. Figure 3.7 shows trapezoidal membership function. In the scale 0 to 2 indicates absence of cough, 1 to 4 indicates low cough and 3 above is considered as high cough. These ranges of absence, low, and high have been fixed based on observance of cough levels in patients and existing perception.

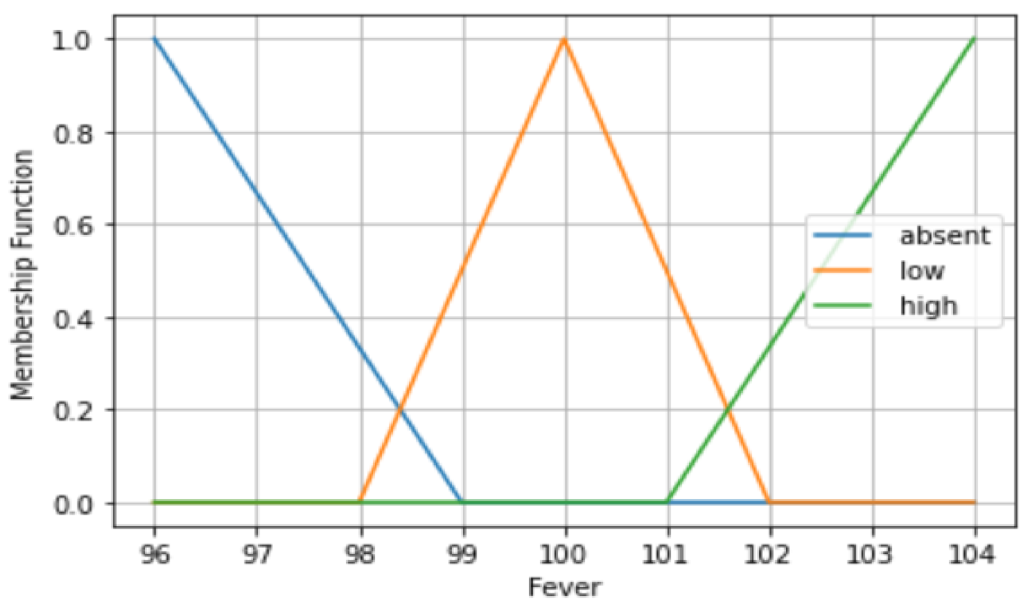


Figure 3.6: Triangular membership function for fever

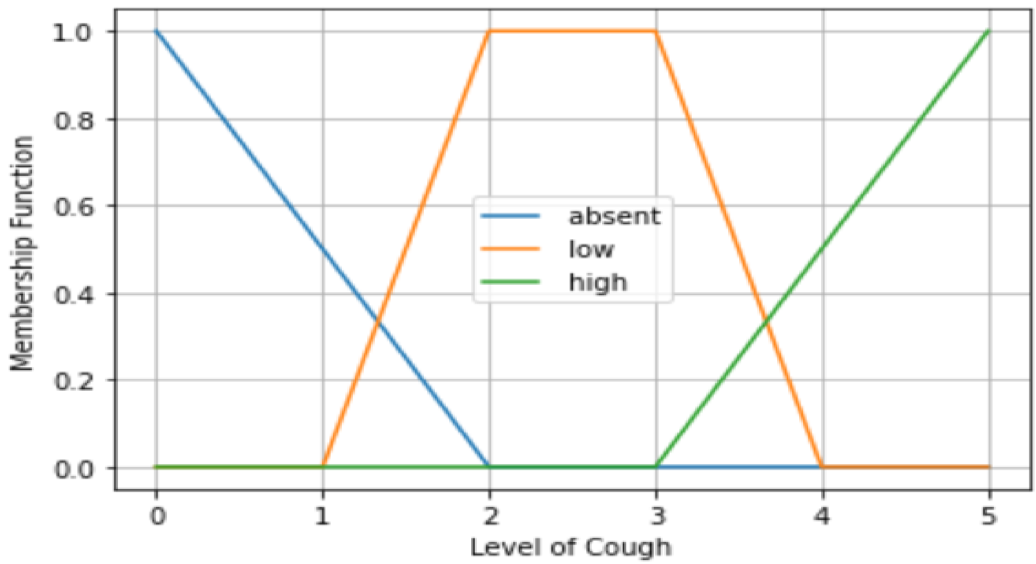

Figure 3.7: Trapezoidal membership function for cough

### 3.3.3. Respiratory distress

Respiratory distress is a critical identifying feature in the context of COVID-19, reflecting the severity of the viral impact on the respiratory system. For Respiratory distress too trapezoidal membership function has been used as shown in Figure 3.5. In the figure it is seen that 0 to 2 is considered as absence of respiratory distress, 1 to 4 is considered as low respiratory distress and 3 to 5 is the high respiratory distress scenario. Here too the ranges of absence, low, and high have been fixed based on observance of cough levels in patients and existing perception.

Since trapezoids represent range or interval better therefore trapezoidal membership function has been chosen for respiratory distress. However, triangular membership function could have been implemented here as well. Mathematical representation of triangular and trapezoidal membership function is given in section 3.8.

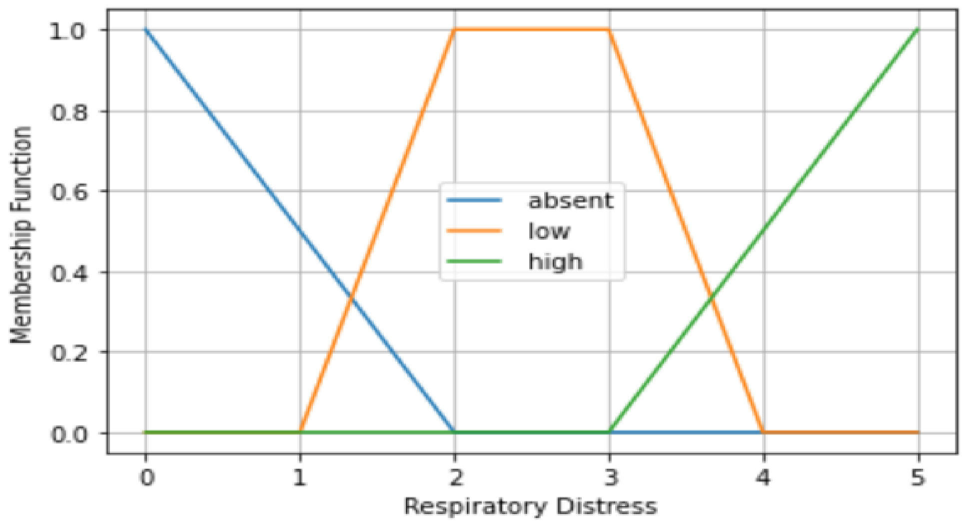


Figure 3.8: Trapezoidal membership function for respiratory distress

### 3.3.4. Loss of smell

Loss of smell, or anosmia, is another notable identifying feature for COVID-19. Individuals infected with the virus commonly report a sudden and unexplained loss of their sense of smell. Loss of smell has been fuzzified using singleton membership function as represented in Figure 3.9. Singleton membership function has value 1 for only one member (when loss of smell=1) and for rest it is 0. Mathematical representation of singleton membership function is given in Section 3.1.

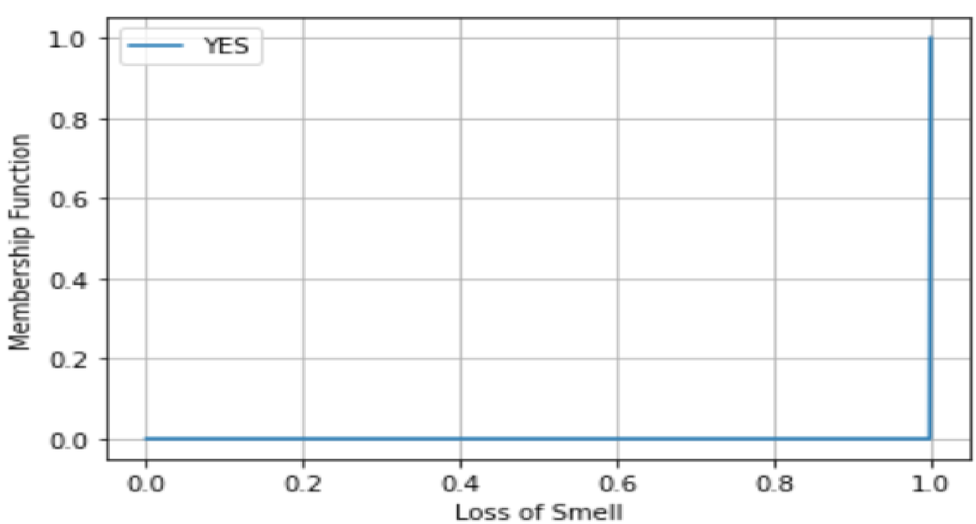


Figure 3.9: Singleton membership function for loss of smell

#### 3.3.5. Tastelessness

The loss of taste, or ageusia, is another crucial feature for identifying COVID-19, often accompanying the related symptom of anosmia, or loss of smell. This peculiar symptom has become recognized as a key marker of COVID-19 infection. Singleton membership function is chosen for tastelessness as shown in Figure 3.10. This singleton membership function has value 1 when tastelessness=1.

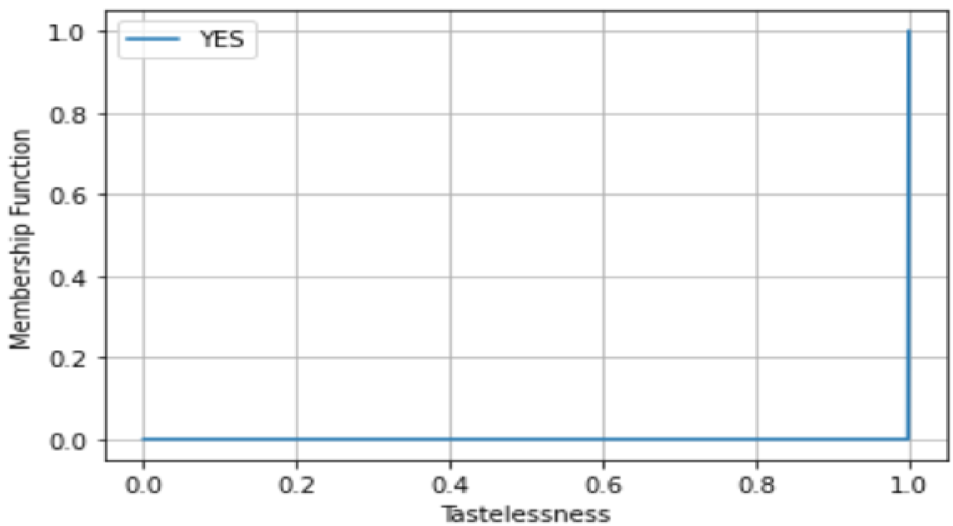


Figure 3.10: Singleton membership function for tastelessness

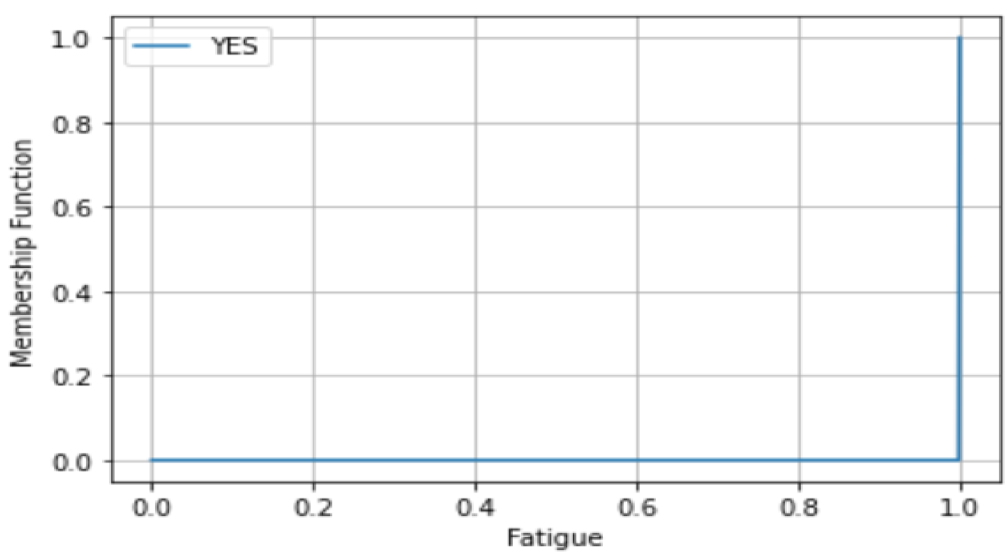


Figure 3.11: Singleton membership function for fatigue

### 3.3.6. Fatigue

Individuals infected with the virus commonly report an overwhelming sense of exhaustion and weariness, often persisting for an extended period. Unlike general tiredness, COVID-19-related fatigue tends to be more profound and can occur even in individuals with mild symptoms. Singleton membership function is chosen for fatigue as shown in Figure 3.11. This singleton membership function has value 1 when fatigue is present and 0 otherwise.

### 3.3.7. Headache

Individuals infected with COVID-19 often report experiencing headaches, which may range from mild to severe. Singleton membership function is chosen for headache as shown in Figure 3.12. This singleton membership function has value 1 when headache is present and 0 otherwise.

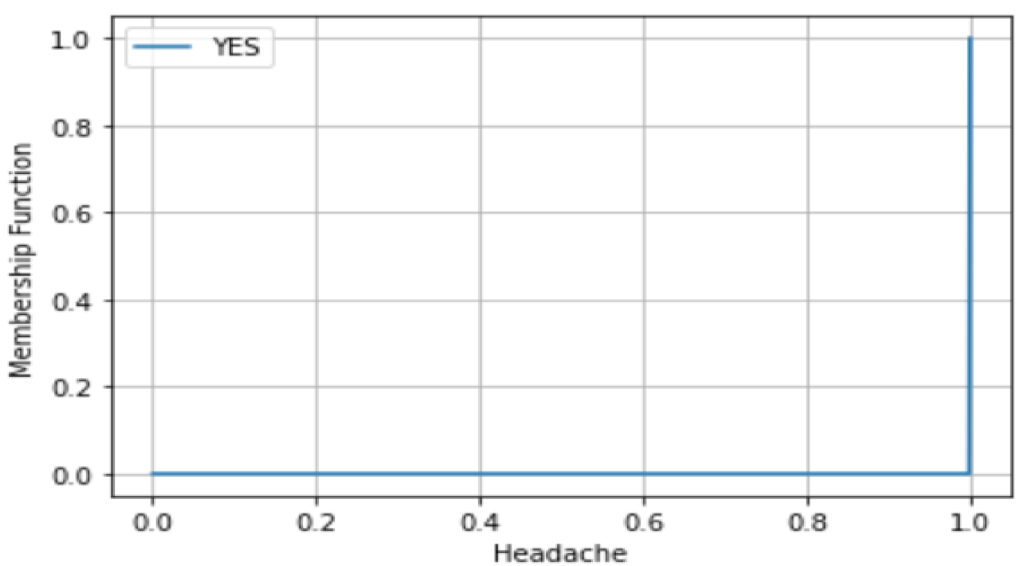


Figure 3.12: Singleton membership function for headache

### 3.3.8. Diarrhea

While respiratory symptoms are predominant, some individuals infected with the virus may experience gastrointestinal symptoms, including diarrhea. For input of diarrhea, singleton membership function is used as represented in Figure 3.13. This singleton membership function has value 1 when diarrhea is observed.

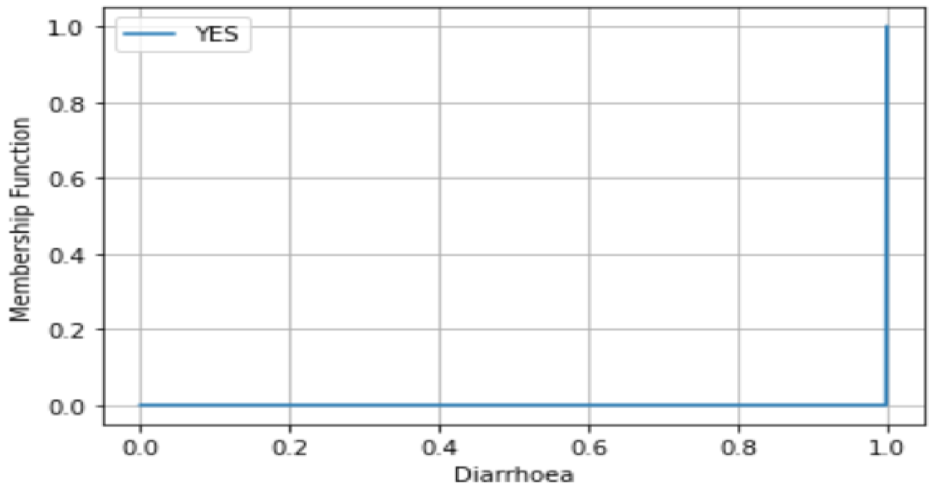


Figure 3.13: Singleton membership function for diarrhea

### 3.3.9. Body Joint pain

Joint pain, or arthralgia, has been identified as a less common but noteworthy feature of COVID-19. Some individuals infected with the virus may experience discomfort and pain in their joints. For input of body or joint pain, singleton membership function is used as represented in section 3.14. This singleton membership function has value 1 when body or joint pain=1.

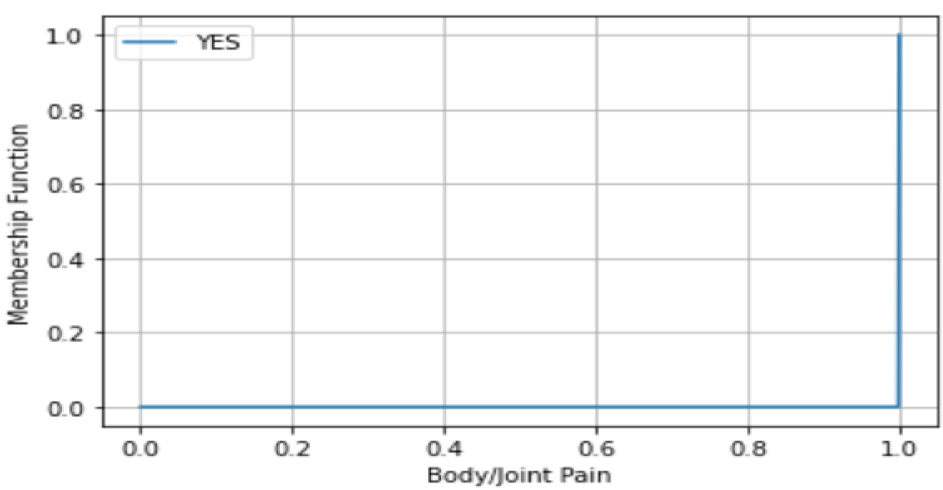


Figure 3.14: Singleton membership function for body/joint pain

### 3.4. Defining membership function for output

Triangular membership function is used for output as shown in Figure 3.15. The shape of the membership function determines how smooth or sharp the transition is from one membership value to another, and how much overlap or gap there is between adjacent fuzzy sets. While trapezoids are good for range, it also has relatively slower transitions. Incase of COVID-19 possibility deduction and providing recommendation, much carefulness was needed to be maintained as many decisions depend on the outcome. Therefore, a sharp transition was needed, and triangular membership function was selected. 0 to 3 is considered a state where test is not required. 2 to 5 is the state where isolation is needed and 5 beyond requires to be tested immediately. As this virus spreads fast, to make it safer for the users, 5 was taken as the threshold point to stay untested. These ranges were decided upon based on existing observed perceptions. The equation for triangular membership function is discussed in Section 3.1

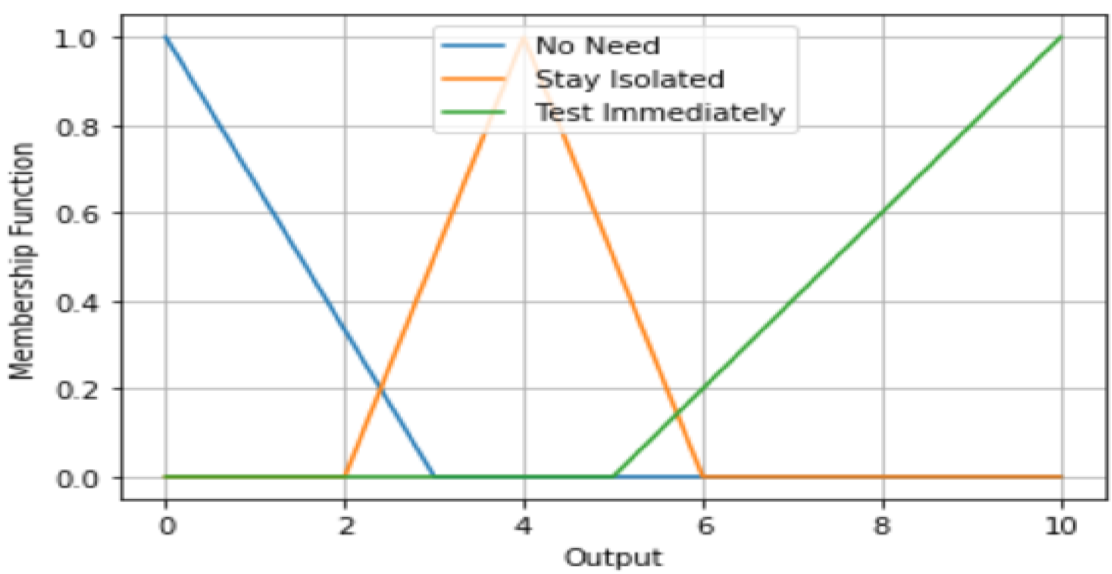


Figure 3.15: Triangular membership function for output

### 3.5. Defining Fuzzy ruleset

To determine the fuzzy ruleset based on which the system will perform, initially a rule table has been prepared with all combination of the input variables mapped to a particular output.

The table consists of total 1,728 combinations. For ease of understanding an instance of the table is presented in Table 3.1

Table 3.1: An instance of rule-based table

| Fever | Level of cough | Respiratory Distress | Tastelessness | Fatigue | Headache | Joint / Body pain | Loss of Smell | Diarrhea | OUTPUT |
|---|---|---|---|---|---|---|---|---|---|
| ABSENT | ABSENT | ABSENT | Y | Y | Y | Y | Y | Y | Test Immediately |
| ABSENT | ABSENT | ABSENT | Y | Y | Y | Y | Y | N | Test Immediately |
| ABSENT | ABSENT | ABSENT | Y | Y | Y | Y | N | Y | Test Immediately |
| ABSENT | ABSENT | ABSENT | Y | Y | Y | Y | N | N | Test Immediately |
| ABSENT | ABSENT | ABSENT | Y | Y | Y | N | Y | Y | Test Immediately |
| ABSENT | ABSENT | ABSENT | Y | Y | Y | N | Y | N | Test Immediately |
| ABSENT | ABSENT | ABSENT | Y | Y | Y | N | N | Y | Test Immediately |
| ABSENT | ABSENT | ABSENT | Y | Y | Y | N | N | N | Test Immediately |
| ABSENT | ABSENT | ABSENT | Y | Y | N | Y | Y | Y | Test Immediately |
| ABSENT | ABSENT | ABSENT | Y | Y | N | Y | Y | N | Test Immediately |
| ABSENT | ABSENT | ABSENT | Y | Y | N | Y | N | Y | Test Immediately |
| ABSENT | ABSENT | ABSENT | Y | Y | N | Y | N | N | Test Immediately |
| ABSENT | ABSENT | ABSENT | Y | Y | N | N | Y | Y | Test Immediately |
| ABSENT | ABSENT | ABSENT | Y | Y | N | N | Y | N | Test Immediately |
| ABSENT | ABSENT | ABSENT | Y | Y | N | N | N | Y | Test Immediately |

From this rule base table finally 7 rules have been deduced, using fuzzy AND operation and fuzzy OR operation between the inputs. Fuzzy AND operation denotes minimum between two values while the OR operation denotes maximum between two values. The deduced rules are mentioned below:

Rule 1: IF Tastelessness=1 or Loss of smell=1 THEN Test Immediately.

Rule 2: IF Respiratory Distress= Low OR High THEN Test Immediately.

Rule 3: IF Fever=Absent AND Level of Cough=Absent AND Respiratory Distress=Absent AND Tastelessness=0 AND Loss of smell=0 THEN Not Needed.

Rule 4: IF (Fever=Low OR High) AND Level of Cough=Absent AND Respiratory Distress=Absent AND Headache=N THEN Stay Isolated.

Rule 5: IF Fever=Absent AND Level of Cough=Low AND Respiratory Distress=Absent AND Headache=N THEN Stay Isolated.

Rule 6: IF (Fever=Low OR High) AND (Level of Cough=Low OR High) AND (Respiratory Distress= Low OR High) THEN Test Immediately.

Rule 7: IF (Fever=Low OR High) AND (Level of Cough=Low OR High) THEN Test Immediately.

For this project seven rules have been deduced to prepare the ruleset. Through this ruleset most commonly prevalent scenarios are addressed. The intention was to keep the ruleset simple so that it is easier to understand, implement and maintain. Besides, with fewer rules, computational overhead decreases and provides faster response time, which is much needed in case of implementation dealing with medical diagnosis. Avoiding overfitting with too many rules and achieving better interpretability were also points of concern during preparation of the rule set. However, the design of the system is kept simple and flexible to accommodate more symptoms and rules as different variants of the virus brings in new dimensions.

### 3.6. Different Defuzzification Techniques

Defuzzification is a crucial step in fuzzy logic systems, converting fuzzy output sets into crisp values for further decision-making or control. Here are some common defuzzification techniques (Wang & Chen, 2014) discussed in brief:

**COA method:**

This COA or center of area method is also known as centroid method and is one of the popular defuzzification techniques.

- COA method calculates the center of area of the fuzzy output set, weighted by the membership values.
- Let μ(x) represent the membership function of the fuzzy output set.
- COA is calculated as:

$$\boldsymbol{COA} = \frac{\int x \cdot \mu(x)\, dx}{\int \mu(x)\, dx} \quad \text{.......................................................} (3.5)$$

**BOA method:**

This BOA or bisector of area method is used if the requirement is symmetrical output membership function.

- BOA method finds the point on the x-axis where the area under the membership function on each side of the point is equal.
- Let $x_1$ and $x_r$ represent the left and right bounds of the fuzzy output set.
- BOA is calculated as:

$$\boldsymbol{BOA = \frac{x_1 + x_r}{2}} \quad \text{.......................................................... (3.6)}$$

**MOM method:**

The MOM or mean of maximum method gives the input with the largest membership value as output.

- MOM method calculates the centroid of the maximum membership values.
- Let argmax($\mu(x)$) represent the set of points where $\mu(x)$ attains its maximum value.
- MOM is calculated as:

$$\boldsymbol{MOM = \frac{1}{|argmax(\mu(x))|} \sum xi \in argmax\left(\mu(x)\right)^{xi}} \quad \text{............................ (3.7)}$$

**SOM method:**

This SOM or smallest of maximum method chooses the smallest value from the maximum output of membership function.

- SOM method selects the crisp output value corresponding to the maximum membership value.
- Let max($\mu(x)$) represent the maximum membership value.
- SOM is calculated as:

$$SOM = \text{argmax}(\mu(x)) \quad \text{.......................................................... (3.8)}$$

**LOM method:**

This LOM or Largest of maximum method chooses the maximum value from the maximum output of membership function.

- LOM method selects the crisp output value corresponding to the maximum membership value, but it chooses the largest one if there are multiple maxima.
- LOM is calculated as:

$$LOM = \max\{x|\mu(x)=\max(\mu(x))\} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3.9)$$

The different defuzzification techniques are graphically represented below in Figure 3.16.

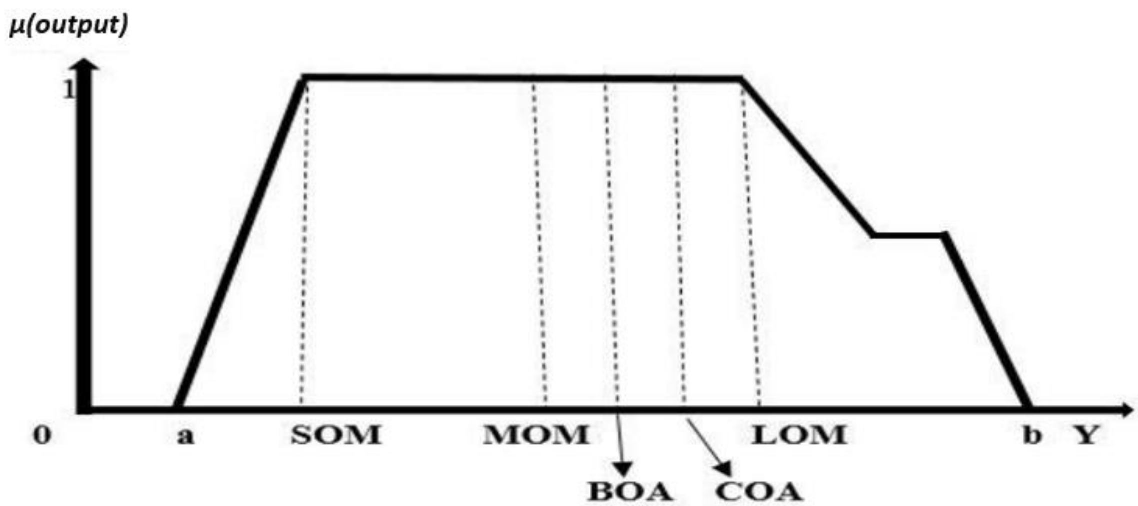


Figure 3.16: Different defuzzification techniques

## 3.7. Summary

While navigating the complexities of COVID-19 prediction, the significance of a well-structured FIS architecture becomes evident in its ability to handle uncertainty. The comprehensive exploration of the system architecture of the Fuzzy Inference System (FIS) for predicting COVID-19 has provided valuable insights into the intricate design considerations and methodologies crucial for developing effective predictive models. By dissecting the various components, such as input variables, linguistic rules, fuzzy inference engine, and defuzzification methods, the foundation for a robust and adaptable system has been laid.

# CHAPTER 4
# SYSTEM SETUP AND IMPLEMENTATION

This pivotal chapter delves into the intricate details of the system setup and implementation of a Fuzzy Inference System (FIS) designed to forecast the dynamics of the COVID-19 pandemic. As the world grapples with the complexities of this global health crisis, the integration of advanced computational methodologies becomes imperative for enhancing the predictive capabilities and facilitating more informed decision-making. This chapter unfolds the behind-the-scenes intricacies of the FIS, unraveling the meticulous steps taken in its setup and implementation to harness the potential of fuzzy logic in forecasting the spread and impact of COVID-19.

## 4.1. System Requirements and Tools Used

The successful implementation of this Fuzzy Inference System (FIS) necessitated the utilization of specific hardware and software elements tailored to this project's requirements. The system requirements and tools employed are outlined below:

**Hardware Requirements:**

- Standard computer systems with sufficient processing power and memory to execute the FIS algorithm efficiently.
- Stable internet connectivity to support web application deployment and user interaction.

**Software Requirements:**

- Python (version 3.8.12): Chosen for its versatility and extensive collection of libraries suitable for machine learning-based system implementation.
- Scikit-Fuzzy and PyIT2FLS: Python toolboxes utilized for fuzzy algorithm implementation, providing robust functionalities for handling fuzzy logic operations and linguistic variable definitions.

- PHP: Selected as the scripting language for web development due to its flexibility and seamless integration with HTML.
- MySQL Database Server: Employed for database implementation, chosen for its scalability and reliability in handling data storage and retrieval operations.

## 4.2. Simulation and Fuzzy System Implementation

The simulation and implementation of the Fuzzy Inference System (FIS) involved a series of systematic steps aimed at configuring and deploying the system within this application tool. The following steps were followed:

**Fuzzy Logic Algorithm Design:**

The first step in this project is creating a fuzzy logic algorithm. This means making a set of instructions that can understand things like how severe a cough is or how tired someone feels. This algorithm is expected to help predict how likely someone has COVID-19 based on their symptoms. So, the symptoms were carefully chosen like temperature, respiratory troubles or coughing, and then made into categories for how severe these symptoms are felt, like "low," "medium," or "high." Finally, rule set was created to help the algorithm make decisions. These rules help it understand the symptoms and make predictions about COVID-19.

**Integration of Fuzzy System into Application Tool:**

After carefully designing the fuzzy logic algorithm, the next step is to smoothly combine it with the application tool. Using Python, which is a flexible and strong programming language, the fuzzy logic algorithm was written. Special libraries like Scikit-Fuzzy and PyIT2FLS were used, that have many features designed for working with fuzzy systems. By coding and integrating the algorithm carefully, it was ensured that it works well with the application tool. This makes it possible to predict things in real-time and show results smoothly, making the application tool easier and more useful to use.

**Web Application Development:**

While integrating the fuzzy logic algorithm, the focus was also to create a user-friendly web interface. This interface was designed to be easy to use while effectively fulfilling the purpose. A combination of PHP, HTML, and CSS was used to build it. The goal is to make it simple for users to input their symptoms and get accurate predictions about COVID-19 likelihood. By using HTML to structure the elements, CSS to make them look good, and PHP for handling behind-the-scenes tasks, a web interface was created that is dynamic and responsive. This not only makes it more engaging for users but also ensures that people with different needs can easily use the application tool.

## 4.3. Login/Signup Page:

The login page of this portal serves as the secure gateway, facilitating user to access their own account on registering with necessary information. Designed with a user-centric approach, this interface stands as the first point of interaction, where individuals are authenticated and granted access to their respective accounts. Featuring an intuitive layout, this login page prioritizes simplicity without compromising on security. This page as shown in Figure: 4.1 provides user prompt to sign-up or login to the system. Signup section is given for new users who are interested to enroll to the Covid calculator portal for the first time and it is completed by providing email id, name and password. Login section is given for existing user to be authenticated with email id and password. Option for retrieving password is also provided with the “Forgot your password” section.

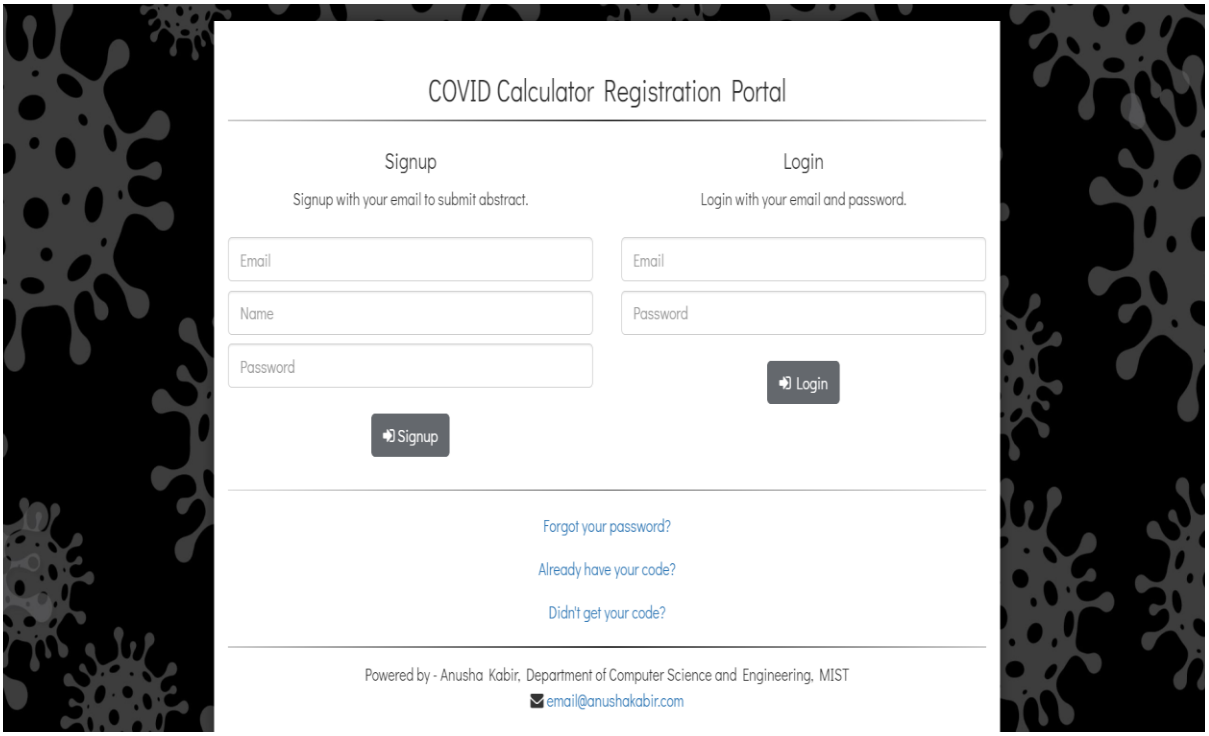


Figure 4.1: Login/Signup page

## 4.4. Input collection and result display page

This page as shown in Figure: 4.2 is divided into two sections.

- The first section on the left side of the page is dedicated for input collection from user. There are nine input fields for the nine symptoms: fever, cough, respiratory distress, tastelessness, fatigue, headache, body-pain, loss of smell and diarrhea which were mentioned as identifying factors for COVID-19. And a submit button which leads to the output.
- The second section on the right is dedicated for displaying results. After the inputs are provided and submit button is pressed, result appear on the right-side table in a descending order of date and time. Result is stored in table firstly in the form of score out of 10 and secondly in terms of explanation of the relevant score.

Let's observe an example below in Figure: 4.3 of how the input collection page collects input from user and results to output. Here the user inputs 99 degrees fever, a cough level of 3, minimal respiratory distress of level 1, followed by a ‘Yes’ for tastelessness and for the rest of the 5 features: fatigue, headache, body pain, loss of smell and diarrhea ‘No’ has been chosen as input as shown in Table: 4.1. On submitting this obtained input, the result board changes with the newest result on top. In this case the result is shown in Figure 4.4. The zone marked with blue rectangle is the latest result obtained.

The result shows a value 8 on a scale of 10 and is explained as possessing high possibility of being covid positive.

Table 4.1: Sample Input

| **Fever** | **Cough** | **Respiratory Distress** | **Tasteless -ness** | **Fatigue** | **Head -ache** | **Body pain** | **Loss of Smell** | **Diarrhea** |
|---|---|---|---|---|---|---|---|---|
| 99 | 3 | 1 | Yes | No | No | No | No | No |

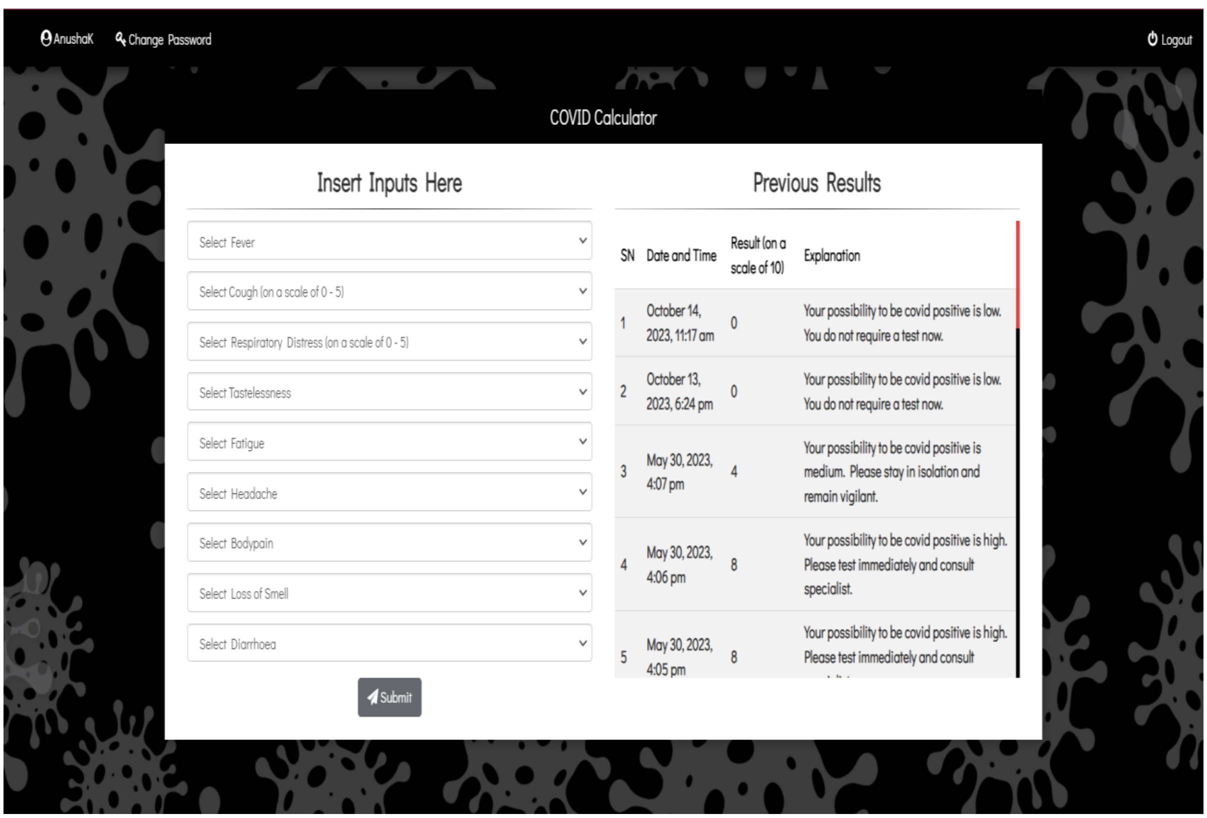


Figure 4.2: Input collection and result display page

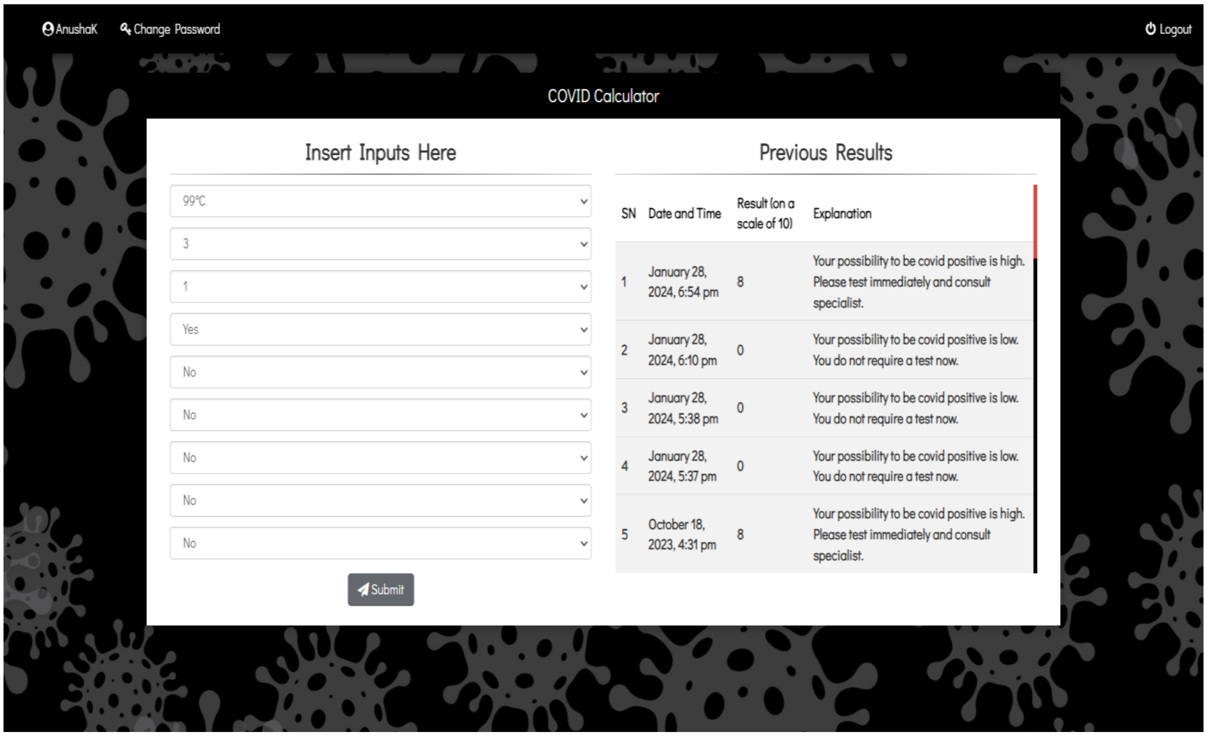


Figure 4.3: Smple Input collection

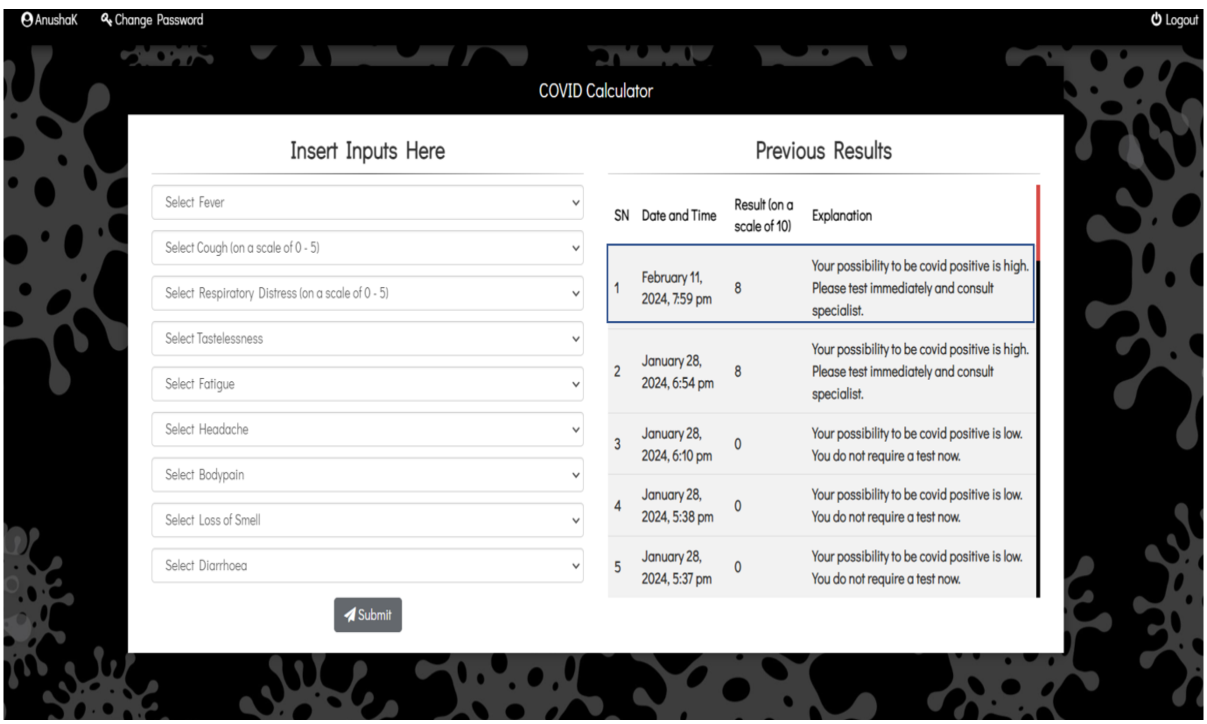


Figure 4.4: Sample Result Display

### 4.5. Summary

In summary, this chapter has undertaken a thorough exploration of the complex procedures involved in establishing, executing, and refining a user-friendly web Graphical User Interface (GUI) system. This journey commenced with a meticulous evaluation of the most suitable hardware and software elements, meticulously chosen to harmonize effortlessly with the specific demands and objectives of the project at hand. By meticulously configuring and seamlessly integrating these components, this study has effectively laid a sturdy groundwork for the forthcoming phases of execution and development.

# CHAPTER 5
# RESULT ANALYSIS AND DISCUSSION

Understanding the data collected and interpreting its implications are pivotal aspects of any research endeavor. In this chapter, the comprehensive analysis of the results obtained from the study is presented, along with an examination of the patterns, trends, and relationships that emerge. Through rigorous analysis and critical discussion, the significance of the findings and their implications for both theoretical understanding and practical applications are aimed to be elucidated, while also highlighting opportunities for future research exploration, theory, practice, and future research directions.

## 5.1. Scenario Overview and Symptom Combinations

To gain insights into the most effective defuzzification technique for deriving results, the analysis delved into several scenarios. These scenarios were carefully crafted to encompass a range of symptom combinations, allowing to assess the performance of different techniques across various contexts. Exploring the sample scenarios outlined in Table 5.1 provides valuable insights into the diverse symptom combinations considered.

Table 5.1: Scenario Overview and Symptom Combinations

| Sl | Fever | Cough | Respira -tory Distress | Taste- lessness | Fati -gue | Head -ache | Body pain | Loss of Smell | Diarr -hea |
|---|---|---|---|---|---|---|---|---|---|
| Scenario-1 | 100 | 1 | 0 | No | No | No | No | No | No |
| Scenario-2 | 100 | 0 | 0 | Yes | No | No | No | No | No |
| Scenario3 | 102 | 2 | 1 | No | No | No | No | No | Yes |
| Scenario-4 | 101 | 0 | 0 | Yes | Yes | Yes | No | No | No |
| Scenario-5 | 99 | 0 | 0 | No | No | No | No | Yes | No |

Subsequently, the analysis of outcomes generated by various defuzzification techniques, as illustrated in Table 5.2, sheds light on their respective effectiveness. This comparative analysis will offer a nuanced understanding of each technique's ability to generate accurate predictions based on the presented scenarios, aiding in determining the most suitable approach for the predictive model.

Table 5.2: Comparative Analysis of Defuzzification Techniques

| Sl | COA | BOA | MOM | SOM | LOM |
|---|---|---|---|---|---|
| Scenario-1 | 4 | 4 | 4 | 3 | 5 |
| Scenario-2 | 6 | 5 | 4 | 5 | 6 |
| Scenario3 | 8 | 7 | 5 | 8 | 7 |
| Scenario-4 | 9 | 9 | 6 | 6 | 8 |
| Scenario-5 | 8 | 5 | 7 | 6 | 5 |

### 5.1.1. Scenario 1:

In Scenario 1, as mentioned in Table: 5.1, the individual is perceived with a fever intensity of 100 and a mild cough level of 1, without experiencing any other respiratory distress or symptoms. Notably, the absence of symptoms such as tastelessness, fatigue, headache, body pain, loss of smell, and diarrhea is observed. Following defuzzification, the outputs obtained through various techniques indicate moderate likelihoods of COVID-19 infection, ranging from 3 to 5 on a scale of 10. The Center of Area (COA) method assigns a value of 4 as shown in Table: 5.2, suggesting a cautious approach such as isolation and vigilance. Figure 5.1 illustrates the output distribution for Scenario 1

### 5.1.2. Scenario 2:

Scenario 2 from Table: 5.1 portrays an individual with a fever intensity of 100 and no significant coughing, but with the presence of tastelessness as the sole symptom. Other symptoms such as fatigue, headache, body pain, loss of smell, and diarrhea are absent. Upon defuzzification, the outputs generated as shown in Table: 5.2 range from 4 to 6, with the COA method assigning a value of 6, indicating a relatively higher likelihood of COVID-19 infection. This scenario warrants prompt isolation and testing due to the presence of a specific symptom associated with the virus. Figure 5.2 illustrates the output distribution for Scenario 2.

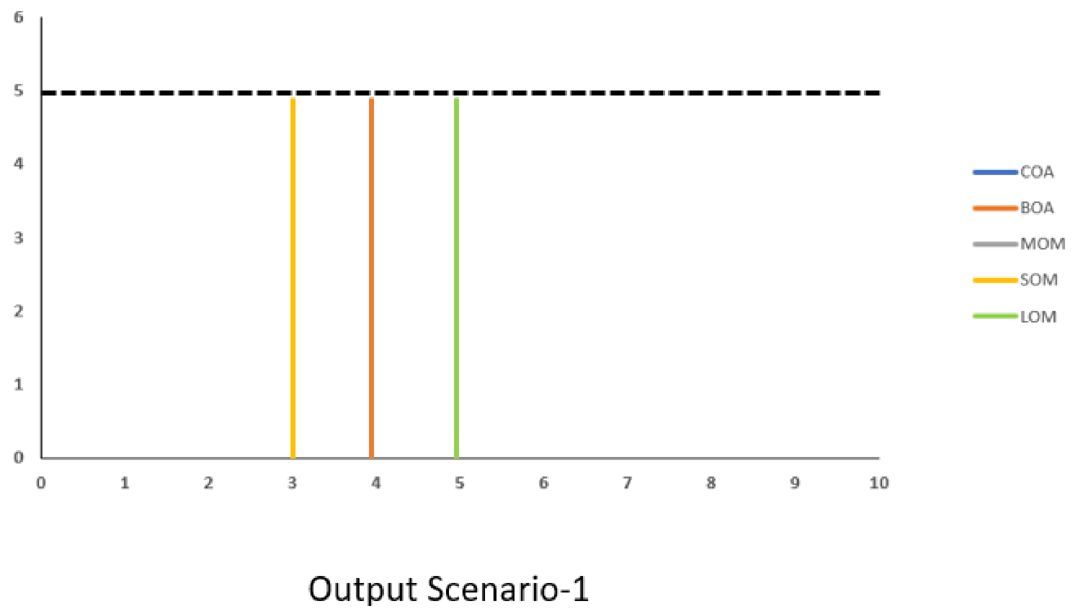


Figure 5.1: Output Distribution for Scenario 1

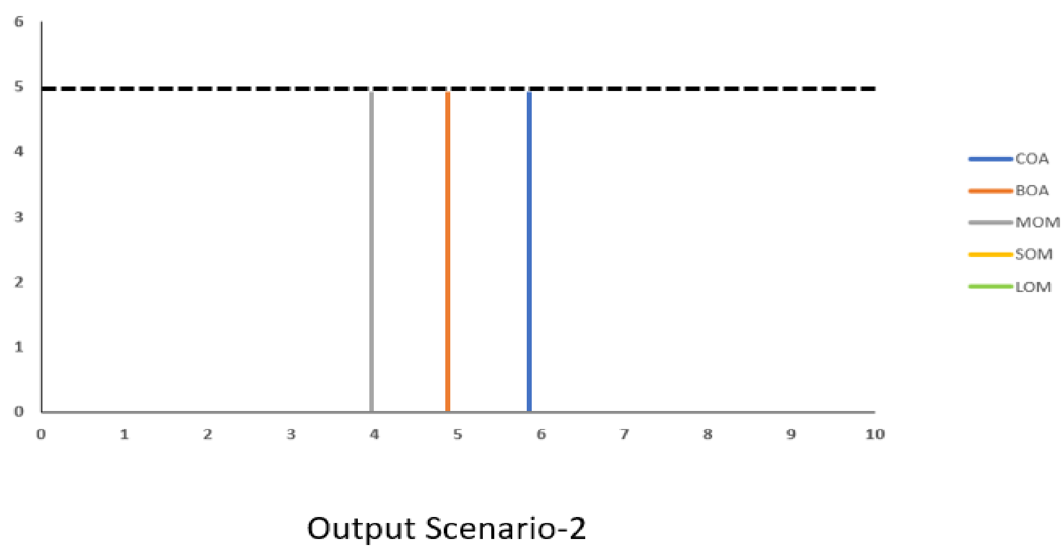


Figure 5.2: Output Distribution for Scenario 2

### 5.1.3. Scenario 3:

In Scenario 3, the individual presents with a higher fever intensity of 102 and a more pronounced cough level of 2, along with additional symptoms such as respiratory distress and diarrhea. Despite the absence of tastelessness, fatigue, headache, body pain, and loss of smell, the presence of respiratory distress and diarrhea elevates the overall risk. Following defuzzification, the outputs range from 5 to 8, with the COA method assigning a value of 8, indicating a significant likelihood of COVID-19 infection. Immediate isolation and testing are recommended in this scenario to mitigate potential transmission and ensure prompt medical intervention. Figure 5.3 illustrates the output distribution for Scenario 3.

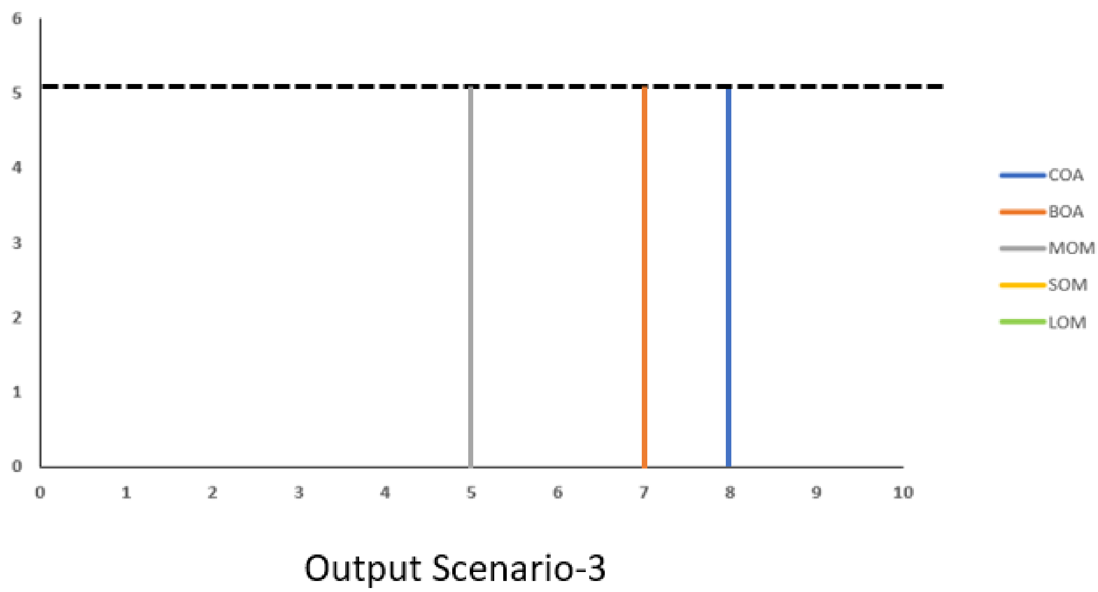


Figure 5.3: Output Distribution for Scenario 3

### 5.1.4. Scenario 4:

In Scenario 4, from Table: 5.1, the individual is shown to have a fever intensity of 101 and a cough level of 0, with no respiratory distress, body pain, loss of smell and diarrhea. However, tastelessness, fatigue, headache was observed. Following different defuzzification techniques as shown in Table: 5.2, the outputs range from 6 to 9, with the COA method assigning a value of 9, indicating a significant likelihood of COVID-19 infection. Immediate isolation and testing are recommended in this scenario to mitigate potential transmission and ensure prompt medical intervention. Figure 5.4 illustrates the output distribution for Scenario 4.

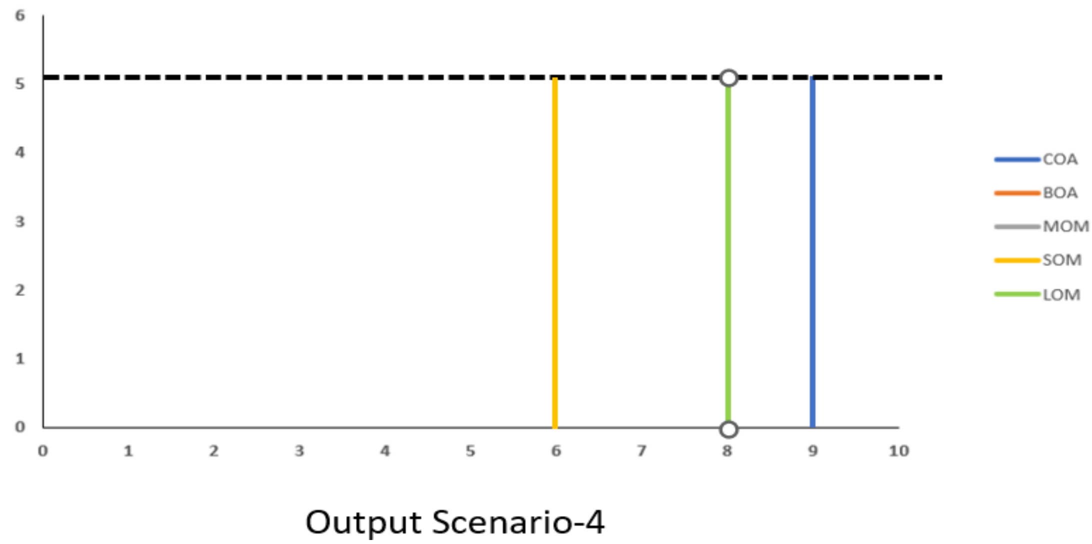


Figure 5.4: Output Distribution for Scenario 4

### 5.1.5. Scenario 5:

In Scenario 5, the individual is presented with a fever intensity of 99 and loss of smell while all other symptoms are absent. Following defuzzification, the outputs range from 5 to 8, with the COA method assigning a value of 8, indicating a significant likelihood of COVID-19 infection. Immediate isolation and testing are recommended in this scenario to mitigate potential transmission and ensure prompt medical intervention. Figure 5.5 illustrates the output distribution for Scenario 5.

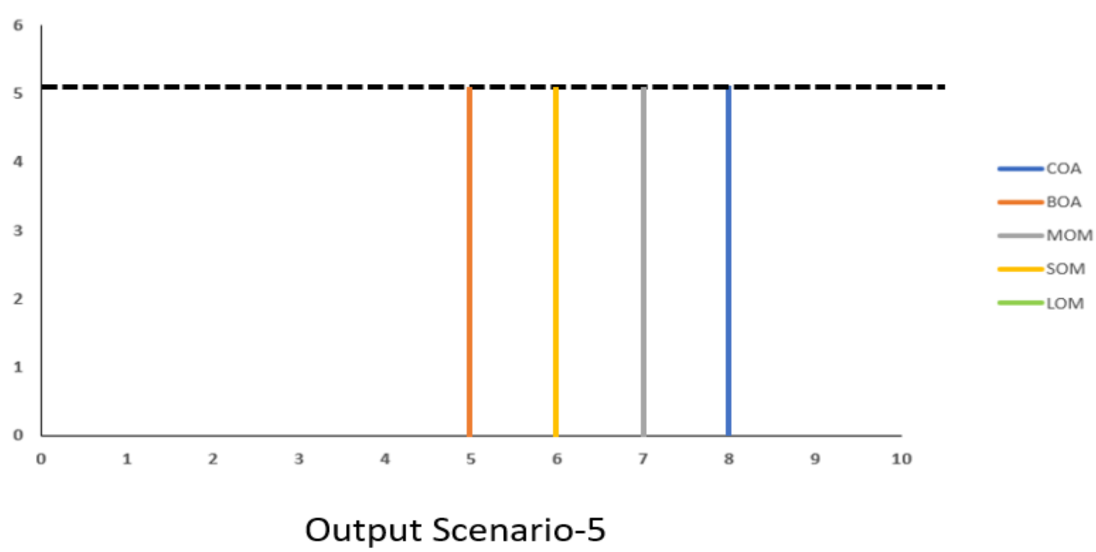


Figure 5.5: Output Distribution for Scenario 5

After thorough observation and evaluation, it became apparent that the COA method consistently yielded results closest to the expected values. By effectively determining the center of the fuzzy set, COA provides a crisp output that closely aligns with the anticipated outcomes. Hence, it was decided to leverage the precise and reliable output generated by the COA method for communicating results to users.

## 5.2. Summary:

In contexts such as Bangladesh, where addressing COVID-19 remains a challenge due to widespread reluctance and ignorance toward testing, our system holds significant potential. By leveraging data-driven insights and decision-making support, our system can contribute to raising awareness among the populace. Early diagnosis facilitated by our system can lead to timely interventions, ultimately reducing the incidence of severe cases and mortality rates associated with COVID-19.

# CHAPTER 6
# CONCLUSION

## 6.1. Outcomes

The outcomes of the developed system are:

- A fuzzy inference system for early prediction of Coronavirus disease.
- Comparative assessment of the result and validation of the proposed system.
- A web application tool designed with fuzzy inference system as backend.

In conclusion, the research has demonstrated the efficacy of using fuzzy logic models in predicting the probability of COVID-19 positivity based on a range of clinical and demographic variables. The findings underscore the potential of fuzzy logic approaches to enhance diagnostic accuracy and facilitate early detection of COVID-19 cases.

## 6.2. Contributions

### 6.2.1. Technical Contribution:

This research culminated in the development of a sophisticated web tool utilizing a Fuzzy Inference System (FIS) to predict COVID-19 infection risk based on an individual's symptoms and demographics. The implementation of advanced computational methodologies enabled the creation of a user-friendly interface, facilitating seamless interaction and accessibility for users of diverse backgrounds. Through meticulous testing and validation, the tool has demonstrated its efficacy in accurately identifying individuals at high risk of COVID-19 infection, thereby contributing to improved diagnostic capabilities in the field of healthcare.

### 6.2.2. Social Contribution:

The deployment of the developed web tool addresses a pressing societal need by providing a reliable platform for early detection and intervention of COVID-19 cases. By enabling individuals to assess their risk of infection based on symptom profiles and demographics, the tool empowers users to seek timely medical attention, potentially leading to better health outcomes. This proactive approach to healthcare management not only promotes individual well-being but also contributes to the collective effort in containing the spread of the virus within communities.

### 6.2.3. Economical Contribution:

The utilization of the FIS-based web tool presents significant economic benefits by optimizing resource allocation in healthcare systems. By facilitating early identification of individuals at high risk of COVID-19 infection, the tool minimizes unnecessary healthcare expenditures associated with delayed diagnosis and treatment. Moreover, the enhanced accuracy of the FIS in diagnosing COVID-19, considering the uncertainty of symptoms, reduces the likelihood of misdiagnosis and subsequent resource wastage. This streamlined approach to healthcare delivery contributes to cost savings and efficiency gains, ultimately benefiting healthcare providers, insurers, and healthcare systems as a whole

## 6.3. Limitations

While the study marks a significant milestone in the application of fuzzy logic to COVID-19 prediction, it is imperative to acknowledge its inherent limitations.

Primarily, for the sake of project simplicity, a selection was made to focus on the nine most critical symptoms of COVID-19. However, future endeavors will undoubtedly seek to expand this scope by incorporating additional contributing symptoms to enhance the comprehensiveness of the predictive model.

Moreover, in a concerted effort to elevate the accuracy and precision of the system, the integration of neural networks alongside fuzzy logic methodologies is deemed essential. Additionally, it is crucial to recognize that the reliance on retrospective data and the specific demographics of the study population may impose constraints on the generalizability of our findings.

Therefore, forthcoming research endeavors should prioritize the conduct of larger-scale studies encompassing diverse demographic cohorts, alongside the incorporation of real-time data collection methodologies. By embracing these measures, future models can aspire to achieve a heightened level of accuracy and efficacy in COVID-19 prediction and management.

### 6.4. Future Works

Despite these limitations, this research contributes valuable insights to the field of COVID-19 diagnosis and management. By leveraging fuzzy logic models, healthcare practitioners may be better equipped to identify and prioritize individuals at higher risk of COVID-19 infection, thereby optimizing resource allocation and improving patient outcomes.

Moving forward, further exploration of fuzzy logic-based prediction models in healthcare settings is recommended, with a focus on refining model parameters, validating findings across different populations, and integrating predictive analytics into clinical practice.

By embracing interdisciplinary collaboration and leveraging advances in computational methods, the effort continues to innovate in the fight against COVID-19 and other emerging infectious diseases.

In summary, this research highlights the importance of using data-focused methods for diagnosing and managing COVID-19. By employing advanced techniques like fuzzy logic and predictive modeling, healthcare workers can better tackle the complex challenges of this pandemic. These tools give them the information they need to make smart decisions, plan interventions, and lessen the impact on public health during this global crisis.

# REFERENCE

Abdalla, A. N., Ali, K., Paw, J. K. S., Rifai, D., & Faraj, M. A. (2018). A novel Eddy Current testing error compensation technique based on Mamdani-Type Fuzzy coupled differential and absolute probes. Sensors, 18(7), 2108. https://doi.org/10.3390/s18072108

Araújo, V. J. S., Guimarães, A. J., De Campos Souza, P. V., Rezende, T. S., & Araújo, V. S. (2019). Using Resistin, glucose, age and BMI and pruning fuzzy neural network for the construction of expert systems in the prediction of breast cancer. Machine Learning and Knowledge Extraction, 1(1), 466–482. https://doi.org/10.3390/make1010028

Asl, A. a. S., Ershadi, M. M., & Sotudian, S. (2021). Fuzzy Expert Systems for Prediction of ICU Admission in Patients with COVID-19. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2104.12868

Bangladesh: WHO Coronavirus Disease (COVID-19) dashboard with vaccination data. (n.d.-b). WHO Coronavirus (COVID-19) Dashboard with Vaccination Data. https://covid19.who.int/region/searo/country/bd

Castillo, O., & Melín, P. (2020). Forecasting of COVID-19 time series for countries in the world based on a hybrid approach combining the fractal dimension and fuzzy logic. Chaos, Solitons & Fractals, 140, 110242. https://doi.org/10.1016/j.chaos.2020.110242

Cihan, P. (2020). Fuzzy Rule-Based system for predicting daily case in COVID-19 outbreak. 2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT). https://doi.org/10.1109/ismsit50672.2020.9254714

Erraguntla, M., Zapletal, J., & Lawley, M. (2017b). Framework for Infectious Disease Analysis: A comprehensive and integrative multi-modeling approach to disease prediction and management. Health Informatics Journal, 25(4), 1170–1187. https://doi.org/10.1177/1460458217747112

Fatima, S. A., Hussain, N., Balouch, A., Rustam, I., Saleem, M., & Asif, M. (2020). IoT enabled Smart Monitoring of Coronavirus empowered with Fuzzy Inference System. International Journal of Advance Research, Ideas and Innovations in Technology, 6(1), 188–194.

Gadekallu, T. R., & Khare, N. (2016). An Efficient System for Heart Disease Prediction Using Hybrid OFBAT with Rule-Based Fuzzy Logic Model. Journal of Circuits, Systems, and Computers, 26(04), 1750061. https://doi.org/10.1142/s021812661750061x

Jaekel, J., Mikut, R., & Bretthauer, G. (2004). Fuzzy control systems. ResearchGate. https://www.researchgate.net/publication/258484987_Fuzzy_Control_Systems

Kanimozhi, U., Ganapathy, S., Manjula, D., & Kannan, A. (2018). An intelligent risk prediction system for breast cancer using fuzzy temporal rules. National Academy Science Letters, 42(3), 227–232. https://doi.org/10.1007/s40009-018-0732-0

Kaur, A., Bhardwaj, A., & Been, U. A. H. (2014). Genetic neuro fuzzy system for hypertension diagnosis. Heart, 19, 25.

Kazeminezhad, M. H., Etemad-Shahidi, A., & Mousavi, S. J. (2005). Application of fuzzy inference system in the prediction of wave parameters. Ocean Engineering, 32(14–15), 1709–1725. https://doi.org/10.1016/j.oceaneng.2005.02.001

Khamparia, A., Jain, R., Rani, P., Gupta, D., Khanna, A., & Castill, O. (2021). An adaptive neuro fuzzy modelling and prediction system for diagnosis of COVID-19. Applied and Computational Mathematics, 124-139.

Omoregbe, N. A., Ndaman, I. O., Misra, S., Abayomi-Alli, O., & Damaševičius, R. (2020). Text Messaging-Based medical diagnosis using natural language processing and fuzzy logic. Journal of Healthcare Engineering, 2020, 1–14. https://doi.org/10.1155/2020/8839524

Onari, M. A., Yousefi, S., Rabieepour, M., Alizadeh, A., & Rezaee, M. J. (2021). A medical decision support system for predicting the severity level of COVID-19. Complex & Intelligent Systems, 7(4), 2037–2051. https://doi.org/10.1007/s40747-021-00312-1

Pathak, A. K., & Arul Valan, J. (2019). A Predictive Model for Heart Disease Diagnosis Using Fuzzy Logic and Decision Tree. Advances in Intelligent Systems and Computing, 131–140. https://doi.org/10.1007/978-981- 13-9680-9_10

Sharma, S., Madaan, V., Agrawal, P., & Garg, N. K. (2018b). Heart disease prediction using Fuzzy system. In Communications in computer and information science (pp. 424–434). https://doi.org/10.1007/978-981-13-3140-4_38

Sharma, P., & Saxena, K. (2017). Application of fuzzy logic and genetic algorithm in heart disease risk level prediction. International Journal of Systems Assurance Engineering and Management, 8(S2), 1109–1125. https://doi.org/10.1007/s13198-017-0578-8

Tandra, S., Gupta, D., Amudha, J., & Sharma, K. (2020). A Fuzzy-Neuro-Based clinical decision support system for disease diagnosis using symptom severity. In Advances in intelligent systems and computing. https://doi.org/10.1007/978-981-15-2475-2_9

Thiyagarajan, M. (2017). Hybrid Neuro-Fuzzy system for prediction of lung diseases based on the observed symptom values. ResearchGate. https://www.researchgate.net/publication/318946565_Hybrid_Neuro-Fuzzy_System_for_Prediction_of_Lung_Diseases_Based_on_the_Observed_Symptom_Values

Vinarti, R. A., & Hederman, L. (2019). A personalized infectious disease risk prediction system. Expert Systems with Applications, 131, 266–274. https://doi.org/10.1016/j.eswa.2019.04.042

Wang, Y., & Chen, Y. (2014). A comparison of Mamdani and Sugeno fuzzy inference systems for traffic flow prediction. Journal of Computers, 9(1). https://doi.org/10.4304/jcp.9.1.12-21

Wulandari, R. D., Surarso, B., Irawanto, B., & Farikhin. (2021). Predict recovery risk rate of COVID-19. Journal of Physics: Conference Series, 1943(1), 012122. https://doi.org/10.1088/1742-6596/1943/1/012122

Zhu, H., Li, W., & Niu, P. (2020). The novel coronavirus outbreak in Wuhan, China. Global Health Research and Policy, 5(1). https://doi.org/10.1186/s41256-020-00135-6

**APPENDIX-A**

**Project Implementation Code:**

In this project, numpy, pandas, sklearn has been used. Below is the Python code for the project. Defining membership function for both input and output, rule set aggregation and defuzzification, all are done using this Python code.

```
Import numpy as np
From numpy import linspace
From numpy import multiply

Def centroid(x, mfx):

   Sum_moment_area = 0.0
   Sum_area = 0.0

   # If the membership function is a singleton fuzzy set:
   If len(x) == 1:
      Return (x[0] * mfx[0] / np.fmax(mfx[0], np.finfo(float).eps).astype(float))

   # else return the sum of moment*area/sum of area
   For i in range(1, len(x)):
      X1 = x[i – 1]
      X2 = x[i]
      Y1 = mfx[i – 1]
      Y2 = mfx[i]

      # if y1 == y2 == 0.0 or x1==x2: → rectangle of zero height or width
```

```
    If not (y1 == y2 == 0.0 or x1 == x2):
        If y1 == y2:  # rectangle
            Moment = 0.5 * (x1 + x2)
            Area = (x2 – x1) * y1
        Elif y1 == 0.0 and y2 != 0.0:  # triangle, height y2
            Moment = 2.0 / 3.0 * (x2 – x1) + x1
            Area = 0.5 * (x2 – x1) * y2
        Elif y2 == 0.0 and y1 != 0.0:  # triangle, height y1
            Moment = 1.0 / 3.0 * (x2 – x1) + x1
            Area = 0.5 * (x2 – x1) * y1
        Else:
            Moment = ((2.0 / 3.0 * (x2 – x1) * (y2 + 0.5 * y1))
                      / (y1 + y2) + x1)
            Area = 0.5 * (x2 – x1) * (y1 + y2)

        Sum_moment_area += moment * area
        Sum_area += area

    Return (sum_moment_area / np.fmax(sum_area, np.finfo(float).eps).astype(float))

Def interp_membership(x, xmf, xx, zero_outside_x=True):

    # Not much beats NumPy’s built-in interpolation
    If not zero_outside_x:
        Kwargs = (None, None)
    Else:
        Kwargs = (0.0, 0.0)
    Return np.interp(xx, x, xmf, left=kwargs[0], right=kwargs[1])
```

```
Def singleton_mf(x, params):

    Return multiply(params[1], x == params[0])

Def trapmf(x, abcd):

    Assert len(abcd) == 4, 'abcd parameter must have exactly four elements.'
    A, b, c, d = np.r_[abcd]
    Assert a <= b and b <= c and c <= d, 'abcd requires the four elements \
                                        A <= b <= c <= d.'
    Y = np.ones(len(x))

    Idx = np.nonzero(x <= b)[0]
    Y[idx] = trimf(x[idx], np.r_[a, b, b])

    Idx = np.nonzero(x >= c)[0]
    Y[idx] = trimf(x[idx], np.r_[c, c, d])

    Idx = np.nonzero(x < a)[0]
    Y[idx] = np.zeros(len(idx))

    Idx = np.nonzero(x > d)[0]
    Y[idx] = np.zeros(len(idx))

    Return y

Def trimf(x, abc):
```

```
    Assert len(abc) == 3, 'abc parameter must have exactly three elements.'
    A, b, c = np.r_[abc]     # Zero-indexing in Python
    Assert a <= b and b <= c, 'abc requires the three elements a <= b <= c.'

    Y = np.zeros(len(x))

    # Left side
    If a != b:
        Idx = np.nonzero(np.logical_and(a < x, x < b))[0]
        Y[idx] = (x[idx] – a) / float(b – a)

    # Right side
    If b != c:
        Idx = np.nonzero(np.logical_and(b < x, x < c))[0]
        Y[idx] = (c – x[idx]) / float(c – b)

    Idx = np.nonzero(x == b)
    Y[idx] = 1
    Return y

Def application(environ, start_response):


    Getcon = environ.get('PATH_INFO')
    Getcon = getcon[1:]
    Int_features = getcon.split('-');
```

```
X_fever = np.arange(96, 105, 1)
X_cough = np.arange(0, 6, 1)
X_rd  = np.arange(0, 6, 1)
X_output = np.arange(0, 11, 1)

X_fever = np.arange(96, 105, 1)
X_cough = np.arange(0, 6, 1)
X_rd = np.arange(0,6,1)
X_output= np.arange(0, 11, 1)

#generate fuzzy mf
Fever_abs = trimf(x_fever, [96, 96, 99])
Fever_low = trimf(x_fever, [98, 100, 102])
Fever_high = trimf(x_fever, [101, 104, 104])

Cough_abs = trapmf(x_cough, [0, 0, 0, 2])
Cough_low = trapmf(x_cough, [1, 2, 3, 4])
Cough_high = trapmf(x_cough, [3, 5, 5, 5])

Rd_abs= trapmf(x_rd, [0, 0, 0, 2])
Rd_low = trapmf(x_rd, [1, 2, 3, 4])
Rd_high = trapmf(x_rd, [3, 5, 5, 5])

Output_noneed = trimf(x_output, [0, 0, 3])
Output_isolation = trimf(x_output, [2, 4, 6])
Output_testimm = trimf(x_output, [5, 10, 10])

X_tastelessness = linspace(0, 1, 1001)
```

```
Tastelessness_singleton = singleton_mf(x_tastelessness, [1, 1])

X_fatigue = linspace(0, 1, 1001)
Fatigue_singleton = singleton_mf(x_fatigue, [1, 1])

X_headache = linspace(0, 1, 1001)
Headache_singleton = singleton_mf(x_headache, [1, 1])

X_bodypain = linspace(0, 1, 1001)
Bodypain_singleton = singleton_mf(x_bodypain, [1, 1])

X_losssmell = linspace(0, 1, 1001)
Losssmell_singleton = singleton_mf(x_losssmell, [1, 1])

X_diarrhoea = linspace(0, 1, 1001)
Diarrhoea_singleton = singleton_mf(x_diarrhoea, [1, 1])

In_fever = int_features[0]
In_cough = int_features[1]
In_rd = int_features[2]
In_tastelessness = int_features[3]
In_fatigue = int_features[4]
In_headache = int_features[5]
In_bodypain = int_features[6]
In_losssmell = int_features[7]
In_diarrhoea = int_features[8]

In_fever_abs =  interp_membership(x_fever, fever_abs, in_fever)
```

```
In_fever_low =  interp_membership(x_fever, fever_low, in_fever)
In_fever_high =  interp_membership(x_fever, fever_high, in_fever)

In_cough_abs = interp_membership(x_cough, cough_abs, in_cough)
In_cough_low = interp_membership(x_cough, cough_low, in_cough)
In_cough_high = interp_membership(x_cough, cough_high, in_cough)

In_rd_abs = interp_membership(x_rd, rd_abs, in_rd)
In_rd_low = interp_membership(x_rd, rd_low, in_rd)
In_rd_high = interp_membership(x_rd, rd_high, in_rd)

In_tastelessness_val = interp_membership(x_tastelessness, tastelessness_singleton,
in_tastelessness)

In_tastelessness_no = 0
In_tastelessness_yes = 1
If in_tastelessness_val < 0.5:
    In_tastelessness_no = 1
    In_tastelessness_yes = 0
Else:
    In_tastelessness_no = 0
    In_tastelessness_yes = 1

In_fatigue_val = interp_membership(x_fatigue, fatigue_singleton, in_fatigue)

In_fatigue_no = 0
In_fatigue_yes = 1
If in_fatigue_val < 0.5:
```

```
    In_fatigue_no = 1
    In_fatigue_yes = 0
Else:
    In_fatigue_no = 0
    In_fatigue_yes = 1

In_headache_val = interp_membership(x_headache, headache_singleton, in_headache)

In_headache_no = 0
In_headache_yes = 1
If in_headache_val < 0.5:
    In_headache_no = 1
    In_headache_yes = 0
Else:
    In_headache_no = 0
    In_headache_yes = 1

In_bodypain_val = interp_membership(x_bodypain, bodypain_singleton, in_bodypain)

In_bodypain_no = 0
In_bodypain_yes = 1
If in_bodypain_val < 0.5:
    In_bodypain_no = 1
    In_bodypain_yes = 0
Else:
    In_bodypain_no = 0
    In_bodypain_yes = 1
```

```
In_losssmell_val = interp_membership(x_losssmell, losssmell_singleton, in_losssmell)

In_losssmell_no = 0
In_losssmell_yes = 1
If in_losssmell_val < 0.5:
   In_losssmell_no = 1
   In_losssmell_yes = 0
Else:
   In_losssmell_no = 0
   In_losssmell_yes = 1

In_diarrhoea_val = interp_membership(x_diarrhoea, diarrhoea_singleton, in_diarrhoea)


In_diarrhoea_no = 0
In_diarrhoea_yes = 1
If in_diarrhoea_val < 0.5:
   In_diarrhoea_no = 1
   In_diarrhoea_yes = 0
Else:
   In_diarrhoea_no = 0
   In_diarrhoea_yes = 1


#defining rules
Rule_1 = np.fmax(in_tastelessness_yes,in_losssmell_yes)
Rule_2 = np.fmax(in_rd_low,in_rd_high)
```

```
Rule_3 = np.fmin(np.fmin(np.fmin(in_fever_abs,
np.fmin(in_cough_abs,in_rd_abs)),in_tastelessness_no),in_losssmell_no)

Rule_4 =
np.fmin(np.fmin(np.fmin(np.fmax(in_fever_low,in_fever_high),in_cough_abs),in_rd_abs),in
_headache_no)

Rule_5 =
np.fmin(np.fmin(np.fmin(in_fever_abs,in_cough_low),in_rd_abs),in_headache_no)

Rule_6 =
np.fmin(np.fmin(np.fmax(in_fever_low,in_fever_high),np.fmax(in_cough_low,in_cough_hig
h)),np.fmax(in_rd_low,in_rd_high))

Rule_7 =
np.fmin(np.fmax(in_fever_high,in_fever_low),np.fmax(in_cough_low,in_cough_high))


#rulewise output

Rule_1_2 = np.fmax(rule_1,rule_2)

Rule_1_2_6 = np.fmax(rule_1_2,rule_6)

Rule_1_2_6_7 = np.fmax(rule_1_2_6,rule_7)

Active_output_testimm = np.fmin(rule_1_2_6_7,output_testimm)



Rule_4_5 = np.fmax(rule_4,rule_5)

Active_output_isolation = np.fmin(rule_4_5,output_isolation)



Active_output_noneed = np.fmin(rule_3,output_noneed)


##working with output


Op0 = np.zeros_like(x_output)
```

```
    #aggregating output

    Agg_op =
np.fmax(np.fmax(active_output_testimm,active_output_isolation),active_output_noneed)

    Defuzz_centroid=centroid(x_output, agg_op)

    Output=round(defuzz_centroid)

    Status = ‘200 OK’
    Content = str(output)

    Response_headers = [(‘Content-Type’, ‘text/html’), (‘Content-Length’, str(len(content)))]
    Start_response(status, response_headers)
    Yield content.encode(‘utf8’)
```

The above Python file has been called as an API from Php files.

In this project , for different functionality separate PHP files have been constructed and then called from main PHP file called entry.php. Codes for the entry.php  file is as follows:

```
<?php


require_once('core/controller/define.php');
session_name(s_name);
session_start();

require_once('core/controller/session.php');
require_once('core/controller/table.php');
```

```
require_once('core/controller/page_manager.php');
require_once('core/controller/funcs.php');
require_once('core/model/view.php');
require_once('core/model/insert.php');
require_once('core/model/delete.php');
require_once('core/model/update.php');
require_once('core/controller/routes.php');

session_write_close();

?>
```

**Code for session.php is as follows:**

```
<?php

function is_authenticated(){
	if (isset($_SESSION['auth_var']))return true;
	else return false;
}



function session_authen(){


	$sqlcon = new mysqli(servername, username, password, dbname);
	$sql = $sqlcon->prepare('SELECT id, email, password, name FROM registration
WHERE email=? AND verd="Yes"');
	$sql->bind_param('s', $_POST['email']);
	$sql->execute();
	$sql->bind_result($id, $email, $password, $name);
```

```
	$sql->fetch();
	$sql->close();
	$sqlcon->close();

	if(empty($password))redirect('base', 'info=There is no verified account associated with this email.');
	else{
			$_POST['password'] = $_POST['password'].user_salt;
			$_POST['password'] = hash('sha512', $_POST['password']);
	  if($_POST['password'] === $password){
	    $_SESSION['auth_var'] = generate_random_string();
	    $_SESSION['email'] = $email;
	    $_SESSION['name'] = $name;
	    $_SESSION['id'] = $id;

	    $em = explode('@', $email);

	    $_SESSION['user'] = $em[0];

	    session_regenerate_id(true);

	    redirect('base');
	  }
	  else redirect('base', 'info=Wrong password.');
	}
}
```

```
function signup(){



	$sqlcon = new mysqli(servername, username, password, dbname);
	$sql = $sqlcon->prepare('SELECT * FROM registration WHERE email=?');
	$sql->bind_param('s', $_POST['email']);
	$sql->execute();
	$sql->store_result();
	$count = $sql->num_rows;
	$sql->close();
	$sqlcon->close();

	if($count != 0)redirect('base', 'info=There is already an account associated with this
email.');
	else{



    $vercode = generate_random_small_string();

    $headers = "MIME-Version: 1.0" . "\r\n";
    $headers .= "Content-type:text/html;charset=UTF-8" . "\r\n";
    $headers .= 'From: COVID Calculator<email@anushakabir.com>' . "\r\n";

    $message = '<!DOCTYPE html>
    <html>
	<h2>COVID Calculator</h2>
	<h3>
		Thank you for signing up.
```

```
            </h3>
            <h3 style="font-weight: normal;">
                  Please use '.$vercode.' as your verification code for COVID Calculator
registration.
            </h3>
        </html>';


        mail($_POST['email'], 'COVID Calculator Account Verification', $message, $headers);


        $_POST['password'] = $_POST['password'].user_salt;
        $_POST['password'] = hash('sha512', $_POST['password']);


        $no = 'No';


        $sqlcon = new mysqli(servername, username, password, dbname);
        $sql = $sqlcon->prepare('INSERT INTO registration(email, name, vercode, password,
verd) VALUES(?,?,?,?,?)');
        $sql->bind_param('sssss', $_POST['email'], $_POST['name'], $vercode,
$_POST['password'], $no);
        $sql->execute();
        $sql->close();
        $sqlcon->close();


        redirect('verify/?email='.$_POST['email'].'&info=Please check your email
('.$_POST['email'].') for a verification code. Please also check your spam folder.');


            }


}
```

```
function send_vercode_again(){

	$sqlcon = new mysqli(servername, username, password, dbname);
	$sql = $sqlcon->prepare('SELECT * FROM registration WHERE email=? AND verd="No"');
	$sql->bind_param('s', $_POST['email']);
	$sql->execute();
	$sql->store_result();
	$count = $sql->num_rows;
	$sql->close();
	$sqlcon->close();

	if($count == 0)redirect('verify', 'info=There is no unverified account associated with this email.');
	else{

	$vercode = generate_random_small_string();

	$headers = "MIME-Version: 1.0" . "\r\n";
      $headers .= "Content-type:text/html;charset=UTF-8" . "\r\n";
      $headers .= 'From: COVID Calculator<email@anushakabir.com>' . "\r\n";

	$message = '<!DOCTYPE html>
      <html>
	<h2>COVID Calculator</h2>
	<h3>
		Thank you for signing up.
```

```
            </h3>
            <h3 style="font-weight: normal;">
                    Please use '.$vercode.' as your verification code for COVID Calculator
registration.
            </h3>
        </html>';

        mail($_POST['email'], 'COVID Calculator Account Verification', $message, $headers);

        $no = 'No';

        $sqlcon = new mysqli(servername, username, password, dbname);
        $sql = $sqlcon->prepare('UPDATE registration SET vercode=? WHERE email=?');
        $sql->bind_param('ss', $vercode, $_POST['email']);
        $sql->execute();
        $sql->close();
        $sqlcon->close();

        redirect('verify/?email='.$_POST['email'].'&info=Please check your email
('.$_POST['email'].') for a verification code. Please also check your spam folder.');

            }

}

function submit_vercode(){

    $no = 'No';
```

```
$sqlcon = new mysqli(servername, username, password, dbname);
$sql = $sqlcon->prepare('SELECT vercode, verd FROM registration WHERE email=?');
$sql->bind_param('s', $_POST['email']);
$sql->execute();
$sql->bind_result($vercode, $verd);
$sql->fetch();
$sql->close();
$sqlcon->close();

if(empty($vercode))redirect('verify', 'info=There is no account associated with this email.');
else{
  if($verd == 'Yes')redirect('base', 'info=This account is already verified. Please login.');
  else{
    if($vercode == $_POST['vercode']){

      $yes = 'Yes';

      $sqlcon = new mysqli(servername, username, password, dbname);
      $sql = $sqlcon->prepare('UPDATE registration SET verd=? WHERE email=?');
      $sql->bind_param('ss', $yes, $_POST['email']);
      $sql->execute();
      $sql->close();
      $sqlcon->close();
```

```
                redirect('base', 'info=Verification complete. Please login with your email and password.');

            }
            else{
                redirect('verify/?email='.$_POST['email'].'&info=Please enter your verification code correctly.');
            }
        }
    }
}


function submit_pwreset(){

    $sqlcon = new mysqli(servername, username, password, dbname);
    $sql = $sqlcon->prepare('SELECT * FROM registration WHERE email=? AND verd="Yes"');
    $sql->bind_param('s', $_POST['email']);
    $sql->execute();
    $sql->store_result();
    $count = $sql->num_rows;
    $sql->close();
    $sqlcon->close();

    if($count == 0)redirect('pwreset', 'info=There is no verified account associated with this email.');
    else{
```

```
$new_pw = generate_random_small_string();

$headers = "MIME-Version: 1.0" . "\r\n";
$headers .= "Content-type:text/html;charset=UTF-8" . "\r\n";
$headers .= 'From: COVID Calculator<email@anushakabir.com>' . "\r\n";

$message = '<!DOCTYPE html>
<html>
   <h2>COVID Calculator</h2>
   <h3>
         Thank you for signing up.
   </h3>
   <h3 style="font-weight: normal;">
         Please use '.$new_pw.' as your new password for COVID Calculator
registration.
   </h3>
</html>';

mail($_POST['email'], 'COVID Calculator Password Reset', $message, $headers);



$new_pw = $new_pw.user_salt;
$new_pw = hash('sha512', $new_pw);



$sqlcon = new mysqli(servername, username, password, dbname);
$sql = $sqlcon->prepare('UPDATE registration SET password=? WHERE email=?');
```

```
        $sql->bind_param('ss', $new_pw, $_POST['email']);
        $sql->execute();
        $sql->close();
        $sqlcon->close();

        redirect('base', 'info=Please check your email ('.$_POST['email'].') for a new password.
Please also check your spam folder.');

            }
}

function logout(){
            session_unset();
            session_destroy();
}

?>
```

**Code for table.php is as follows:**

```
<?php

class tr{
    private $contents;

    public function __construct($m = true, $tr_style=null, $tr_class=null){
        if($m){
            $this->contents = '<tr';
```

```
            if(!empty($tr_class))$this->contents .= ' class="'.$tr_class.'"';

            if(!empty($tr_style))$this->contents .= ' style="'.$tr_style.'"';

            $this->contents .= '>';
        }
        else {
            $this->contents = '';
        }

    }

    public function add_cell($cell=null, $colspan=null, $rowspan=null, $align=null,
$style=null, $td_class=null){

        $this->contents .= '<td';

        if(!empty($colspan))$this->contents .= ' colspan="'.$colspan.'"';

        if(!empty($rowspan))$this->contents .= ' rowspan="'.$rowspan.'"';

        if(!empty($align))$this->contents .= ' align="'.$align.'"';

        if(!empty($style))$this->contents .= ' style="'.$style.'"';

        if(!empty($td_class))$this->contents .= ' class="'.$td_class.'"';

        $this->contents .= '>'.$cell.'</td>';
```

```
    }

    public function add_row($tr_class=null, $tr_style=null){
        $this->contents .= '</tr><tr';

        if(!empty($tr_class))$this->contents .= ' class="'.$tr_class.'"';

        if(!empty($tr_style))$this->contents .= ' style="'.$tr_style.'"';

        $this->contents .= '>';
    }

    public function output(){
        return $this->contents.'</tr>';
    }
    public function output2(){
        return $this->contents;
    }
}

?>
```

This way through a combination of Python, PHP, HTML and CSS, the project has been deployed.

**Code for insert.php is as follows:**

```
<?php
```

```
class insert{

    private $fname;

    public function __construct($p1){
        $this->fname = $p1;
    }

    public function fnexec(){

        if($this->fname == 'submit'){

            // if(
            //     !isset($_POST['fever']) ||
            //     empty($_POST['fever']) ||
            //     empty($_POST['cough']) ||
            //     empty($_POST['rd']) ||
            //     empty($_POST['tastelessness']) ||
            //     empty($_POST['fatigue']) ||
            //     empty($_POST['headache']) ||
            //     empty($_POST['bodypain']) ||
            //     empty($_POST['losssmell']) ||
            //     empty($_POST['diarrhoea'])
            //     )redirect('base', 'info=Please enter the required information correctly.');
            // else{
```

```
$params = $_POST['fever'].'-'.$_POST['cough'].'-'.$_POST['rd'].'-'.$_POST['tastelessness'].'-'.$_POST['fatigue'].'-'.$_POST['headache'].'-'.$_POST['bodypain'].'-'.$_POST['losssmell'].'-'.$_POST['diarrhoea'];

$url = 'https://anushakabir.pythonanywhere.com/'.$params;
$ch = curl_init();
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
curl_setopt($ch, CURLOPT_URL, $url);
$result = curl_exec($ch);

$e_date = get_date();


$sqlcon = new mysqli(servername, username, password, dbname);
$sql = $sqlcon->prepare('INSERT INTO covid_data(user_id, e_date, fever, cough, rd, tastelessness, fatigue, headache, bodypain, losssmell, diarrhoea, result) VALUES(?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)');
$sql->bind_param('ssssssssssss', $_SESSION['id'], $e_date, $_POST['fever'], $_POST['cough'], $_POST['rd'], $_POST['tastelessness'], $_POST['fatigue'], $_POST['headache'], $_POST['bodypain'], $_POST['losssmell'], $_POST['diarrhoea'], $result);
$sql->execute();
$sql->close();
$sqlcon->close();


redirect('base', 'info=Result obtained.');
```

```
            // }

        }

    }

}

?>
```

**Code for update.php is as follows:**

```
<?php

class update{

    private $fname;

    public function __construct($p1){
        $this->fname = $p1;
    }

    public function fnexec(){

        if($this->fname == 'update_password'){

            if(!isset($_POST['old_password']) || empty($_POST['old_password']) ||
!isset($_POST['new_password']) || empty($_POST['new_password']) ||
!isset($_POST['new_password_again']) ||
```

```
empty($_POST['new_password_again']))redirect('change_password', 'info=Please enter all the information corrrectly.');
	else{
if($_POST['new_password']!=$_POST['new_password_again'])redirect('change_password', 'info=New passwords do not match.');
		else{
			$sqlcon = new mysqli(servername, username, password, dbname);
			$sql = $sqlcon->prepare('SELECT password FROM registration WHERE id=?');
			$sql->bind_param('s', $_SESSION['id']);
			$sql->execute();
			$sql->bind_result($password);
			$sql->fetch();
			$sql->close();
			$sqlcon->close();

			if(hash('sha512', $_POST['old_password'].user_salt) !== $password)redirect('change_password', 'info=Please enter your old password correctly.');
			else{
				$dat_new_password = hash('sha512', $_POST['new_password'].user_salt);

				$sqlcon = new mysqli(servername, username, password, dbname);
				$sql = $sqlcon->prepare('UPDATE registration SET password=? WHERE id=?');
				$sql->bind_param('ss', $dat_new_password, $_SESSION['id']);
				$sql->execute();
				$sql->close();
				$sqlcon->close();

				redirect('base', 'info=Password changed.');
```

```
                    }

                }

            }

        }

    }

}

?>
```

**Code for delete.php is as follows:**

```
<?php

class delete{

    private $fname;

    public function __construct($p1){

        $this->fname = $p1;

    }
```

```
public function fnexec(){
    if($this->fname == 'del_paper'){
        if(!isset($_GET['p']) || empty($_GET['p']))redirect('base');
        else{
            $sqlcon = new mysqli(servername, username, password, dbname);
            $sql = $sqlcon->prepare('SELECT file, fullpaper_submission_locked FROM
        papers WHERE id=? AND user_id=?');
            $sql->bind_param('ss', $_GET['p'], $_SESSION['id']);
            $sql->execute();
            $sql->store_result();
            $count = $sql->num_rows();
            $sql->execute();
            $sql->bind_result($file, $fullpaper_submission_locked);
            $sql->fetch();
            $sql->close();
            $sqlcon->close();
```

```
    if($count == 0)redirect('base');

    else if($fullpaper_submission_locked == 1)redirect('base', 'info=This paper is
locked.');

    else{

if(!is_dir('/home/mist/cicm2023/abstracts/'.$file))unlink('/home/mist/cicm2023/abstra
cts/'.$file);

            $sqlcon = new mysqli(servername, username, password, dbname);

            $sql = $sqlcon->prepare('DELETE FROM papers WHERE id=? AND
user_id=?');

            $sql->bind_param('ss', $_GET['p'], $_SESSION['id']);

            $sql->execute();

            $sql->close();

            $sqlcon->close();

            redirect('base', 'info=Paper deleted.');
```

```
                }
            }
        }
    }
}
?>
```