Fuzzy-Centric Fog–Cloud Inspired Deep Interval Bi-LSTM Healthcare Framework for Predicting Yellow Fever Outbreak

Prabal Verma, *Member, IEEE*, Tawseef A. Shaikh, *Member, IEEE*, Sandeep K. Sood, Harkiran Kaur, Mohit Kumar, Huaming Wu, *Senior Member, IEEE*, and Sukhpal Singh Gill

***Abstract*—Yellow fever is a vigorous, phlebotomic, vector-borne disease that poses a significant public health threat in regions with high mosquito density and inadequate vaccination coverage. The disease's toxic phase is lethal, making prompt identification and control measures crucial. The emergence of the latest technologies and data analytics techniques, such as edge-cloud computing, data analytics, and machine learning/deep learning, has played a pivotal role in revolutionizing remote healthcare services. Henceforth, applying the abovementioned technologies leads to improvements in the response time, service quality, and location awareness of healthcare systems. Relative to this context, we propose an intelligent fuzzy-centric fog–cloud-assisted healthcare framework to identify and control yellow fever epidemics. Initially, at the fog layer, singular value decomposition is used for data dimensionality reduction analysis and the Fuzzy-C mean clustering (FCM) algorithm is leveraged to get rigorous results. Moreover, for better results and to focus on time-series patterns, the deep interval type 2 fuzzy Bi-LSTM model is proposed at the cloud layer to generate a yellow fever severity index and visualize each yellow fever region based on self-organized maps. In addition, we propose an alert generation mechanism to facilitate real-time decision-making. Finally, results show that the proposed system yields significant efficacy, compared with other state-of-the-art methodologies.**

***Index Terms*—Deep interval fuzzy type-2 Bi-LSTM, fog–cloud computing (CC), Fuzzy-C mean clustering (FCM), self-organized mapping (SOM), singular value decomposition (SVD), temporal mining.**

I. Introduction

YELLOW fever is an acute viral disease attributable to the yellow fever virus, which replicates in lymph nodes and infects dendritic cells and hepatocytes, leading to the eosinophilia degradation of these cells and the release of cytokines [1]. This disease causes 200 000 infections and 30 000 deaths each year [2]. Like other flavivirus infections, yellow fever has no known cure at present. The disease has an incubation period of 3–6 days and two phases: a mild phase and a toxic phase that can be fatal in 20%–50% of the cases.[1] Symptomatic treatment is provided to patients, including paracetamol, aspirin, and antiviral drugs, and hospitalization may be necessary. Symptoms range from mild, e.g., fever, headache, fatigue, loss of appetite, muscle pain, bleeding in the eyes and mouth, and gastrointestinal tract infection [3], [4]. Yellow fever shares some symptoms with malaria, Zika, dengue, and chikungunya viruses, such as high fever, joint pain, muscle pain, headache, fatigue, and nausea [5]. A comparison of yellow fever symptoms with those of malaria, Zika, dengue, and chikungunya viruses is presented in Table I. Here, "###" shows the main symptoms of yellow fever, dengue fever, Zika virus, malaria, and chikungunya virus, and "##" represents symptoms that are not serious but still an important indicator for these diseases, and "#" represents inconsistent symptoms that may or may not be present in a patient from these diseases, and "-" in a person with the same disease represents the absence of symptoms.

The World Health Organization reports that the United States allocates the highest percentage of its gross domestic product toward healthcare expenditures for its citizens [6]. Effective management of healthcare is crucial for the development of any nation. Nowadays, it is essential to use remote detection and monitoring systems to control the outbreak of infectious diseases, as the existing healthcare system has not been able to effectively combat such diseases. The evolution of technology has led to the development of fog computing (FC) and various cloud computing (CC) instances that are easy to set up and can be used in large IT systems [7], [8], [9]. The migration of healthcare

[1][Online]. Available: https://www.who.int/news-room/fact-sheets/detail/yellow-fever

Manuscript received 11 May 2024; accepted 7 June 2024. Date of publication 11 June 2024; date of current version 8 October 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62071327 and in part by Tianjin Science and Technology Program under Grant 22ZYYYJC00020. Recommended by Associate Editor C. Lin. *(Corresponding author: Huaming Wu.)*

Prabal Verma is with the Department of Information Technology, National Institute of Technology Srinagar, Srinagar 190006, India (e-mail: prabalverma357@gmail.com).

Tawseef A. Shaikh is with the Department of Computer Science and Engineering, National Institute of Technology Srinagar, Srinagar 190006, India (e-mail: tawseef.shaikh@nitsri.ac.in).

Sandeep K. Sood is with the Department of Computer Applications, National Institute of Technology Kurukshetra, Kurukshetra 136119, India (e-mail: san1198@gmail.com).

Harkiran Kaur is with the Department of Computer Science, Guru Nanak Dev University, Amritsar 143005, India (e-mail: harkiran1992@gmail.com).

Mohit Kumar is with the Department of Information Technology, National Institute of Technology Jalandhar, Jalandhar 144011, India (e-mail: kumarmohit@nitj.ac.in).

Huaming Wu is with the Center for Applied Mathematics, Tianjin University, Tianjin 300072, China (e-mail: whming@tju.edu.cn).

Sukhpal Singh Gill is with the School of Electronic Engineering and Computer Science, Queen Mary University of London, E1 4NS London, U.K. (e-mail: s.s.gill@qmul.ac.uk).

Digital Object Identifier 10.1109/TFUZZ.2024.3412197

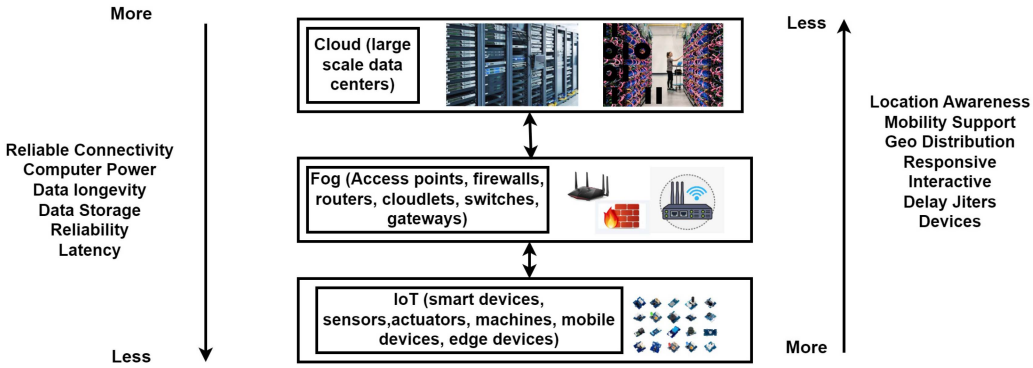

Fig. 1. FC in Healthcare 5.0.

TABLE I
SYMPTOM-BASED COMPARISON OF YELLOW FEVER, CHIKUNGUNYA, ZIKA, DENGUE, AND MALARIA

Symptoms	Chikungunya	Zika Virus	Dengue	Malaria	Yellow Fever
Jaundice	-	-	###	-	###
Abdominal pain	-	-	###	-	###
Red eyes	#	###	-	-	##
Headaches	##	###	###	##	##
Muscle/Joint aches	###	##	##	#	##
Seizures	-	-	-	###	###
Onset post infection	2–7 days	3–12 days	4–7 days	3–6 days	7–30 days
Loss of appetite	-	-	-	-	##
Pain behind eyes	-	#	###	-	-
Nausea	#	#	###	#	##
Decreased urination	-	-	-	-	###
Skin rash	##	###	###	-	-
Itching	#	###	##	-	-
Fatigue	#	###	###	##	##

and other mobile technology applications to the cloud platform has enabled remote real-time services [10]. However, heterogeneous data types over the cloud can lead to latency-sensitive issues, location awareness, and delays in the transmission of Big Data [11]. These factors can result in data transmission dilution, delayed user notification, faulty diagnosis, and potential harm to human life.

A. Motivation and Objectives

To overcome these challenges, a fog layer has been introduced to reduce latency between end users and cloud servers. Motivated by the need for an early warning system, advanced data analysis, real-time monitoring, secure data transmission, and resource optimization, a fog–cloud-assisted secured framework has been developed for predicting and preventing yellow fever outbreaks [12], [13], [14]. By leveraging FC-CC, this framework analyzes data from various sources, such as climate patterns and disease incidence rates, to provide timely alerts and preventive measures. Real-time monitoring of environmental conditions and vector populations enables swift actions to mitigate outbreaks. The framework offers multiple benefits, including secure data transmission, protection of sensitive health information, optimized resource allocation, validation of benefits, accessibility of medical information, and prompt alarm notifications, as shown in Fig. 1. These advantages greatly assist yellow fever-infected individuals and deliver real-time notices to end users within a specified time frame. Moreover, this approach enhances outbreak prediction and prevention strategies, leading to saved lives and improved public health outcomes. The proposed framework addresses the restriction confronted in yellow fever identification and outbreak prevention. Its essential goals are as follows:

1) providing an initial class to users based on their infected symptoms using singular value decomposition (SVD) and Fuzzy-C mean clustering (FCM) algorithm at the fog layer;
2) in-depth analysis of temporal granular series using deep interval type 2 fuzzy Bi-LSTM model at cloud layer;
3) decision making using self-organized mapping (SOM)-based visualization technique;
4) sending instant emergency alert messages to users to prompt timely action.

B. Existing Works

A comprehensive analysis is conducted, comparing various state-of-the-art research articles that center on different diseases. The examination is based on key specifications, including the inclusivity of Internet of Things (IoT), applications domain, CC, FC, prediction model (PM), real-time perspective (RTP), visualization mechanism (VS), and comparison with infectious/other diseases (CID), as detailed in Table II.

C. Article Organization

The rest of this article is organized as follows. Section II focuses on different components of the proposed intelligent healthcare system. Section III comprises a detailed experimental evaluation with dataset generation and results. Finally, Section IV concludes this article with recommendations for future work.

II. PROPOSED SYSTEM

The proposed system, depicted in Fig. 2, is composed of a cloud layer, a middle section (FC), and a front-end section

TABLE II
COMPARISON OF THE PROPOSED APPROACH WITH THE EXISTING WORKS

Reference	Year	Application Domain/Major Contributions	IoT	FC	CC	PM	RTP	VS	CID
[15]	2018	Displayed a novel system for controlling the outbreak of Mosquito-borne diseases	✓	✓	✓	✓	×	×	✓
[16]	2019	Proposed efficient framework to distinguish and control the mosquito-borne diseases	×	✓	✓	×	×	×	✓
[17]	2023	Presented an efficient and novel dengue monitoring framework	✓	✓	✓	✓	×	×	×
[18]	2021	Presented a fog-cloud framework for monitoring and controlling swine flu	×	✓	✓	×	×	×	×
[19]	2022	SDN-enabled Fog nodes specific Early Detection of Communicable Infections (FogCom)	×	✓	×	×	×	×	×
[20]	2022	Demonstrated a novel system to help a person/community related to the irregular screen of COVID-19	✓	✓	✓	✓	×	×	✓
[21]	2021	Showed a novel and energy-efficient methodology for ceaselessly checking the heart for low- power wearable devices.	×	✓	✓	✓	×	×	×
[22]	2023	Displayed an autonomic Edge-assisted Cloud-IoT framework for heart illness prediction	×	✓	✓	✓	×	×	×
[23]	2022	Proposed a novel framework called ABFog which integrates with edge computing gadgets to analyze Heart disease	×	✓	✓	✓	×	✓	×
[24]	2020	Proposed system for identifying individuals affected by dengue at prior stage	×	✓	✓	✓	×	×	×
[25]	2022	Proposed an approach for detecting and tracking Parkinson Disease intensity	×	✓	✓	✓	×	×	×
Proposed system	2023	Presented a novel architecture for yellow fever detection and precaution measures	✓	✓	✓	✓	✓	✓	✓

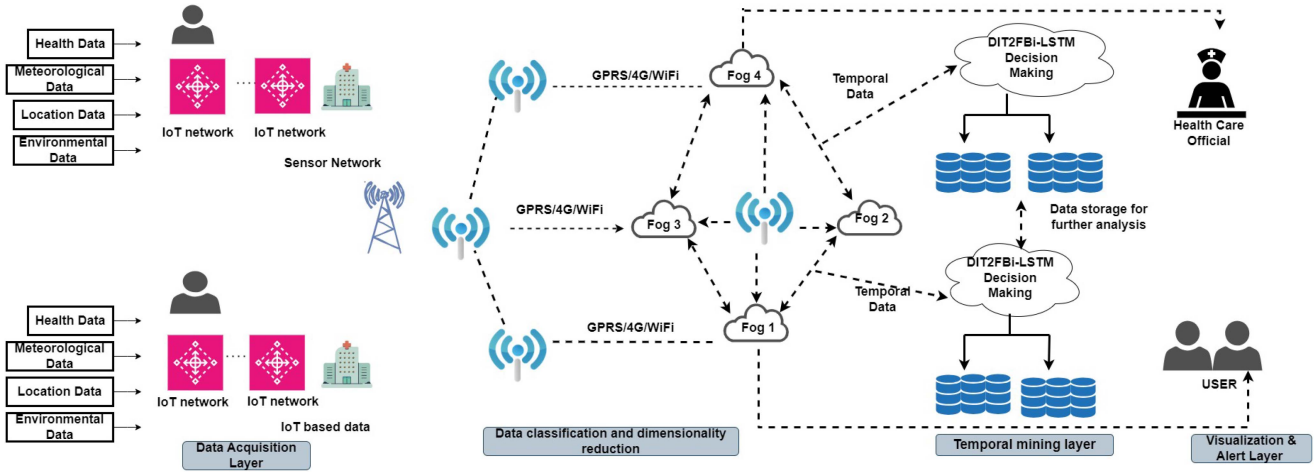

Fig. 2. Proposed framework for yellow fever prediction and mitigation.

(data acquisition layer). The cloud layer section comprises various components, including predictive analysis, SOM-based visualization with Global Positioning System (GPS)-based risk assessment, and health communication. The fog server contains two modules: data collection and dimensionality reduction, and classification using Fuzzy-C mean clustering. The fog layer is connected to the mobile device through General Packet Radio Service communication protocols and also collects real-time data from various acquisition methods and sensors, as illustrated in Fig. 2. The proposed model incorporates a location-based fog server that collects overall data from different devices and transfers it to cloud servers for further decision-making. The data acquisition component is responsible for gathering personal information and yellow fever symptoms from users via the mobile application and IoT network. In addition, environmental sensors are employed to collect useful data related to air quality and other parameters discussed in the fog component layer. The SVD is used for dimensionality reduction and the Fuzzy-C mean clustering algorithm is then utilized to classify infected and uninfected users based on their health symptoms. Moreover, temporal data are transferred to the cloud layer for further processing, prediction, and decision-making. At the cloud layer, deep learning methodology followed by the yellow fever outbreak severity index helps determine the yellow fever outbreak intensity.

A. Fog Component Layer

The proposed model illustrates the use of FC as a communication channel between the cloud and end users. The system security can be enhanced by integrating a cloud access security broker with health fog. FC acts as a bridge between mobile, cloud, and fog servers, extending CC services to edge devices in the network while improving quality of service for real-time applications. Heterogeneous data entered by the user are collected through mosquito sensors at the fog nodes and relayed to the cloud layer for in-depth data processing and analysis. Fog node interaction in a fog layer can cover a city, country, or even a continent. They provide services with low latency, location awareness, and local data processing at the appropriate node to process data in real time [26].

Algorithm 1: Dimensionality Reduction Using SVD.

Input: Dataset DS of dimensions $R * n$ with R as several records and n as total no. of features/attributes for yellow fever.

Output: Reduce the Dataset $(DS)_{R*n}$ to $(DS)_{R*(n-f)}$ as total dimensions discarded with f as total dimensions discarded.

1: Factorize the dataset DS as $(DS)_{R*n} = (P)_{R*R}$ $(Q)_{R*n}(S)^T_{n*n}$.
2: Determine P and Q such that $PP^T = V_R$ and $S^TS = V_n$, here columns of P and S are eigen vectors in orthogonal form of $(DS)(DS)^T$ and $(DS)^T(DS)$, respectively.
3: Calculate $S^-_{(n-f)*(n-f)}$ by removing the last f rows and f columns from S.
4: Calculate $(Q)_{R*n}$ such that Q is diagonal and singular one, and elements in the matrix are non-negative square roots of eigenvalues of P and S in decreasing order.
5: Calculate $Q^-_{R*(n-f)}$ by removing last f columns of Q.
6: Calculate $((DS)_{R*(n-f)}$ as $(DS)_{R*(n-f)} = P_{R*R}$ $Q^-_{R*(n-f)}$ $S^-_{(n-f)*(n-f)}$
7: Exit.

Algorithm 2: Fuzzy-C means clustering Algorithm Working in Our Proposed System.

1: Initially, set the number of clusters as per the dataset and also the value pertaining to the fuzzifier constant. For process termination, set ϵ greater than zero (termination condition).
2: Initialize the membership matrix as $M = [m_{ij}]q * l$, where q is the number of clusters set as optimal for the problem and l is the count of data points.
3: Find the computation of fuzzy cluster centers c_i, for k iterations $c_i^k = \dfrac{\sum_{j=1}^{l}[m_{ij}^{(k-1)}]^n p_j}{\sum_{j=1}^{l}[m_{ij}^{(k-1)}]^n}$
4: Update membership value m_{ij} and c_i^k according to $m_{ij} = \dfrac{1}{\sum_{k=1}^{q}\left(\dfrac{|(p_j - c_i)|}{|(p_j - c_k)|}\right)^{2/n-1}}$
5: If $|M^{(k+1)} - M^{(k)}| > \epsilon$, Stop. Otherwise, go to Step 3 and find membership degrees and new cluster centers until the termination condition is satisfied.

1) Data Collection and Dimensionality Reduction: To start utilizing the framework, health-related data are acquired from IoT devices in real time. The data include a person's health data, meteorological data, location, and environmental data, as shown in Fig. 2. Moreover, to overcome missing data insertion and time synchronization issues because of heterogeneous sensors, universal time synchronization, and expectation maximum methods are used for filling the former objective. Furthermore, the dimensionality reduction method is used to reduce the dimensionality of the data for accurate results. In addition, some vital information can be submitted by users via mobile application. Data values are categorized as follows.

1) *Health related data:* Important parameters given in Table I come under health dataset. Various wearable and bio-sensors are required to collect these data.
2) *Meteorological data:* This dataset encompasses variables, such as rainfall, humidity, and temperature. The data are collected through climate sensors strategically positioned at various locations.
3) *Location data:* This dataset facilitates the identification of infectious and high-risk regions. The utilization of GPS technology enables the acquisition of data across diverse locations and regions.
4) *Environmental data:* Diet quality, surrounding air level, and other risk factors that impact human health are measured. To seamlessly collect these data values, feature extraction techniques, including cross-modality, symbolic aggregation approximation mechanism, and hidden conditional random field extraction method, have been incorporated.

The required number of dimensions can be achieved by removing less efficient and redundant parts, resulting in effective time complexity or lower cost and proper utilization of network parameters. SVD is used for data dimensionality reduction in our proposed framework due to its properties [27]. Algorithm 1 shows the complete steps adopted for dimensionality reduction.

2) Data Classification: The proposed model utilizes fuzzy clustering to generate results. If the output result is "YES" (infected), it updates the user's location on Google Maps, and if it is "NO" (not infected), it sends a precautionary message to the user. Fuzzy logic is a form of set theory and logic that allows predicates to have degrees of applicability rather than just being true or false. Fuzzy sets can potentially have an infinite range of truth values between one and zero [28], [29].

FCM is an effective clustering technique that organizes a dataset into *n* clusters, and each data point is relatively appertained to each cluster. The degree of membership of a data point in a cluster is excessive if it is close to the center of the cluster and low if it is far from the center. FCM iteratively finds the centroid of each cluster and fuzzy pseudopartition until the partition does not change. A set of ten data points, denoted as $Y = \{Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7, Y_8, Y_9, Y_{10}\}$, has been carefully selected according to the specified requirements. These data points correspond to the features presented in Table I (initial symptoms-based classification). The two fuzzy clusters S1 and S2 are taken as subsets for all possible fuzzy subsets of *Y* and are used to classify users by grouping the dataset into these clusters. One includes uninfected users and the other includes potentially infected users. The primary goal of FCM is to minimize the error of the objective function. The mathematical foundation of the objective function is represented as follows:

$$F(M, C, D) = \sum_{i=1}^{2}\sum_{j=1}^{10} m_{ij}^n d_{ij}^2(p_j, c_i) \tag{1}$$

$$F(M,C,D)=\sum_{i=1}^{2}\sum_{j=1}^{10}m_{ij}^{n}|(p_j-c_i)|^2 \tag{2}$$

where $M=m_{ij}$ is the desired partition matrix. Here, $i\in\{1,2\}$ and $j\in\{1,2,3,\ldots,10\}$ represent the membership values of data point j in the ith cluster. The set D represents a collection of data points, and $C=\{c_i\}|_{i=\{1,2\}}$ is defined as the set of cluster centers. In addition, n is the fuzzy robustness parameter that is usually considered for the best statistical results 2 and also controls the fuzziness of the problem. $|p_j-c_i|^2$ is the Euclidean distance determinant. In addition, FCM must meet three conditions to be worth membership $m_{ij}\in[0,1]\forall i,j$.

The submission of the membership values of each data point must be equal to 1

$$\sum_{i=1}^{2}m_{ij}=1\forall j. \tag{3}$$

All membership values in each cluster must be submitted to be less than the number of data points (N)

$$0<\sum_{j=1}^{10}m_{ij}<N. \tag{4}$$

The complete steps for the FCM classifier are presented in Algorithm 2, which is designed to evaluate the category of the user.

3) Temporal Mining Layer: FC provides a feasible platform for ubiquitously sensed data anytime from anywhere. Connectors transmit vital signals, and indicators of yellow fever outbreak prediction accuracy by storing the data in the cloud repository whenever an event happens. The vital signals are stored in an adequate format for analyzing purposes. Since event-triggered mode generates temporal data, it must be represented in the real values format for analysis. Time-series sequence consists of a real value set of different events, which are analyzed using a temporal granulation mechanism [30].

Definition 1 (Temporal mining data pattern): A time-series T can be defined as an ordered list of real-valued variables $T=<t_1,t_2,\ldots,t_n>$, where n is its length. In addition, the dataset will be a long-term series that can be restricted to subsections called subsequences for analysis.

Corollary 1.1 (Time-series pattern): Given a field K, a field time series (FTS) can be defined as a set of m values over a certain period: $<t(s),v(s)>,<t(s+1),v(s+1)>,\ldots,<t(s+m-1),v(s+m-1)>$. Often the values are identified for specific well-defined points in time, in that case, the value may be viewed as a vector $<v(s),v(s+1),\ldots,v(s+m-1)>$.

Corollary 1.2: Given an FTS of various events occurring during the sliding window of δt, temporal granule series at an instance t_a is defined as attributes selected from various datasets to form a relation set A_i. Moreover, the temporal granule formation mechanism is shown in Fig. 3.

Definition 2 (Yellow fever outbreak severity value (YFOSV)): The possibility that a user's health is in an undesirable state at any time instance with the vulnerability for serious consequences

$$YFOSV=\frac{1}{p}(\alpha\times V_1+\beta\times V_2+\gamma\times V_3+\cdots+v\times V_p) \tag{5}$$

where α, β, γ, and v are the respective measures for the pth attribute. A larger YFOSV indicates the user is highly vulnerable to yellow fever. Fig. 4 shows the temporal granularity concept. Temporal granularity helps in calculating YFOSV and provides sensitivity information on different events.

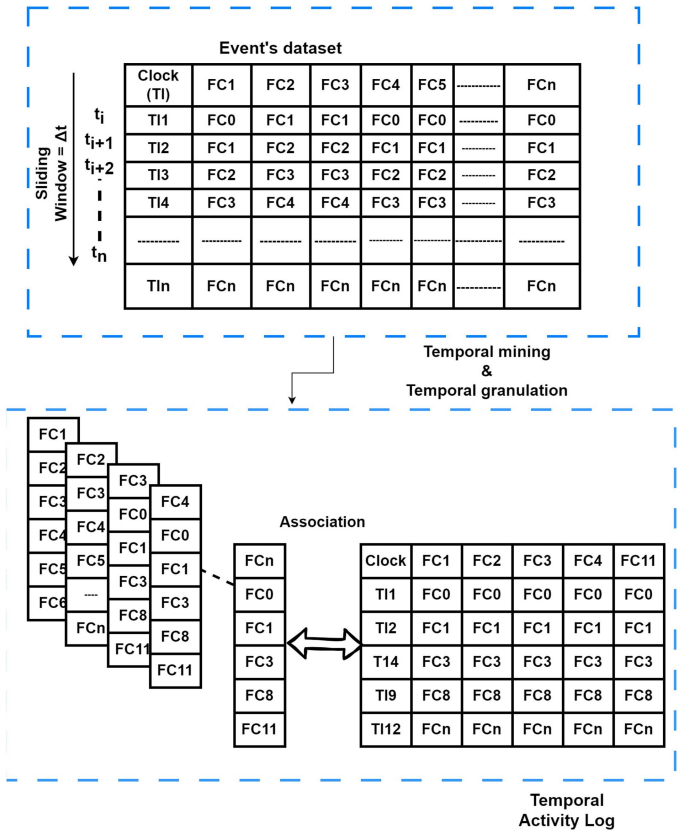

Fig. 3. Temporal granularity mechanism.

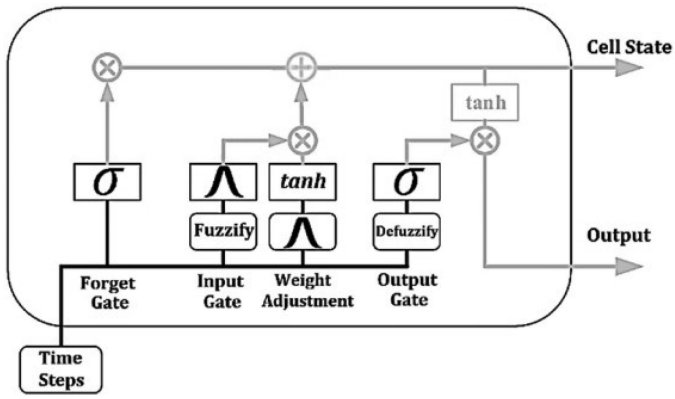

Fig. 4. DIT2FLSTM cell structure.

B. DIT2FBi-LSTM Model

Fig. 4 defines the cell structure of the proposed deep interval type-2 fuzzy long-short-term memory (LSTM) model. The inputs to the DIT2FBi-LSTM model can be represented as i, x_i, and x_{i+1}. The input structure can be represented using Takagi–Sugeno–Kang interval type-2 fuzzy inference as follows:

$$i\left\{\begin{matrix} R_i^{m_1m_2} : \text{if } x_i = F_i^{m_1} \text{ and } x_{i+1} = F_i^{m_2} \\ \text{then } y_i = \left[e_{-i}^{m_1,m_2}, \bar{e}_i^{\bar{m}_1,m_2}\right] \end{matrix}\right\}_{m_1,m_2=1}^{M} \tag{6}$$

where $F_i^{m_1}$ and $F_i^{m_2}$ are, respectively, the interval type-2 fuzzy sets for the input x_i and x_{i+1}, respectively. Moreover, M represents the number of fuzzy rules applied. Other parameters $e_{-i}^{m_1,m_2}$ and $e_i^{-m_1,m_2}$ are the endpoints of a fuzzy rule $R_i^{m_1,m_2}$ or the consequent. We used the product t-norm technique in (6), to determine the firing intervals

$$\begin{cases} \underline{f}^l\left(\mathbf{x}'\right) = \mu_{-x_1^1}\left(x_1'\right) \times \cdots \times \mu_{-x_i'}\left(x_i'\right) \\ \bar{f}^l\left(\mathbf{x}'\right) = \bar{\mu}_{x_1'}\left(x_1'\right) \times \cdots \times \bar{\mu}_{x_i'}\left(\dot{x_i'}\right). \end{cases} \tag{7}$$

The inference part of our proposed model can be entirely portrayed by M-fuzzy rules for the inference process, where the lth rule now has the form

$$R^l : \text{if } x_1 \text{ is } \tilde{F}^l \cdots \text{and } x_i \text{ is } \tilde{F}_i^l, \text{ then } y \text{ is } \tilde{G}^l \quad \forall l = 1, \ldots, M$$

where i is the inputs $x_1 \in X_1, \ldots, x_i \in X_i$ and the output $y \in Y$. The membership function can be formulated as follows:

$$\mu_{R^l}(x, y) = \mu_{A \to \tilde{G}^l}(x, y). \tag{8}$$

The output of each fuzzy rule is $B^l = A_x \circ R^l$, with membership function of $\mu_{B^l}(y)$ as follows:

$$\mu_{B^l}(y) = \bigcup_{x \in X} \left[\mu_{A_x}(x) \cap \mu_{A \to \bar{G}}(x, y)\right] \tag{9}$$

where $\bigcup$ signifies the maximum t-conorm operation [31], and $\circ$ denotes the composition operation. It is pertinent to mention here that $\tilde{F}^l(x^{A'})$ denotes the firing intervals for the fuzzy rule, where $x = x^{A'}$ and $\tilde{F}^l$ is as follows:

$$\tilde{F}^l\left(x'\right) \equiv \left[\underline{f}^l\left(x'\right), \bar{f}^l\left(x'\right)\right]. \tag{10}$$

Using the ruleset and the aggregation of the consequent, the firing output B^l is produced through fuzzy inference [32]

$$B^l : \begin{cases} \text{FOU}\left(B^l\right) = \left[\bar{\mu}_{B'}\left(y \mid x'\right), \bar{\mu}_{B'}\left(y \mid x'\right)\right] \\ \underline{\mu}_{B'}\left(y \mid x'\right) = \underline{f}^l\left(x'\right) \times \underline{\mu}_{G'}(y) \\ \bar{\mu}_{B'}\left(y \mid x'\right) = \bar{f}'\left(x'\right) \times \bar{\mu}_{\bar{G}}(y). \end{cases} \tag{11}$$

Moreover, the final output B^l can be gathered by integrating all rule firing sets B^l on the output. Here, $*$ represents the product ts-norm operation

$$B^l : \begin{cases} \text{FOU}(B) = \left[\underline{\mu}_B\left(y \mid x'\right), \bar{\mu}_B\left(y \mid x'\right)\right] \\ \underline{\mu}_B\left(y \mid x'\right) = \vee \underline{\mu}_{B^M}\left(y \mid x'\right) \\ \bar{\mu}_B\left(y \mid x'\right) = \vee \bar{\mu}_{B^M}\left(y \mid x'\right) \end{cases} \tag{12}$$

where $\vee$ denotes the maximum operation. Then, the type reduced set $I_c(x^{A'})$ is obtained through computing the centroid $C_{B^{A'}}$ of B

$$I_C(x') = C_B(x') = \frac{1}{\left[l_B\left(x'\right), r_B\left(x'\right)\right]} \tag{13}$$

where the two points $l_B(x')$ and $r_B(x')$ are computed through the Karnil-Mendel (K.M) algorithm [33].

C. Prediction and Decision-Making Layer

This layer provides a novel time-series prediction for calculating yellow fever outbreak severity. The bidirectional (DIT2FBi-LSTM) model is used to estimate the severity of YFOSV over the time series of events. Fig. 6 depicts the flow diagram of the proposed decision-making system at the cloud layer. Deep interval type 2 fuzzy Bi-LSTM applies DIT2F-LSTM twice in forward and reverse form, which increases the accuracy of the model and the context available to the algorithm. The forward function of a DIT2FBi-LSTM with M input units and N hidden units can be calculated as

$$d_n^t = \sum_{m=1}^{M} x_m^t w_{mn} + \sum_{n'=1,t>0}^{N} f_{n'}^{t-1} w_{n'n} \tag{14}$$

$$f_n^t = \phi_n(d_n^t) \tag{15}$$

where x^t is the sequential input, and d_n^t represents at time t of unit n the network input to DIT2F-LSTM. f_n^t is the activation function n at time t, w_{mn} is the input weight m toward n, $w_{n'n}$ is the weight of hidden unit n toward n', ϕ_n is the activator function of the hidden layer, and μ is an objective function with p units of output. The backward function can be calculated as

$$\frac{\delta\mu}{\delta w_{np}} = \sum_{t=1}^{T} \frac{\delta\mu}{\delta d_n^t} f_n^t \tag{16}$$

$$\frac{\delta\mu}{\delta d_n^t} = \phi_n'\left(d_n^t\right)\left(\sum_{p=1}^{P} \frac{\delta\mu}{\delta d_n^t} w_{np} + \sum_{n'=1,t>0}^{N} \frac{\delta\mu}{\delta d_{n'}^{t+1}} w_{nn'}\right). \tag{17}$$

Fig. 5 illustrates the integration of DIT2FBi-LSTM into the proposed system. Following the announcement of results, a window of size n is created, containing x label measures. Subsequently, YFOSV is calculated based on these labels, as described in Definition 2 [34], [35].

D. Visualization and Alert Generation Layer

To discard the numbered value and adopt novel methods for visualization, the SOM technique is used [36]. It is an important tool [37] for locating disease hot positions. Moreover, the geographical information system tool is used to analyze the spatial distribution of disease-prone areas and classify yellow fever hot positions using cluster analysis techniques, including Getis-Ord Gi* and SOM. As shown in Fig. 6, mapping is done through ArcGIS 10.2 software, and the SOM technique is also incorporated for the dynamic presentation of the color-coding method to analyze the geographical distribution of yellow fever. The proposed DiT2FBi-LSTM model for prediction is visualized using the U-matrix technique (UMT) [38]. Fig. 6 also

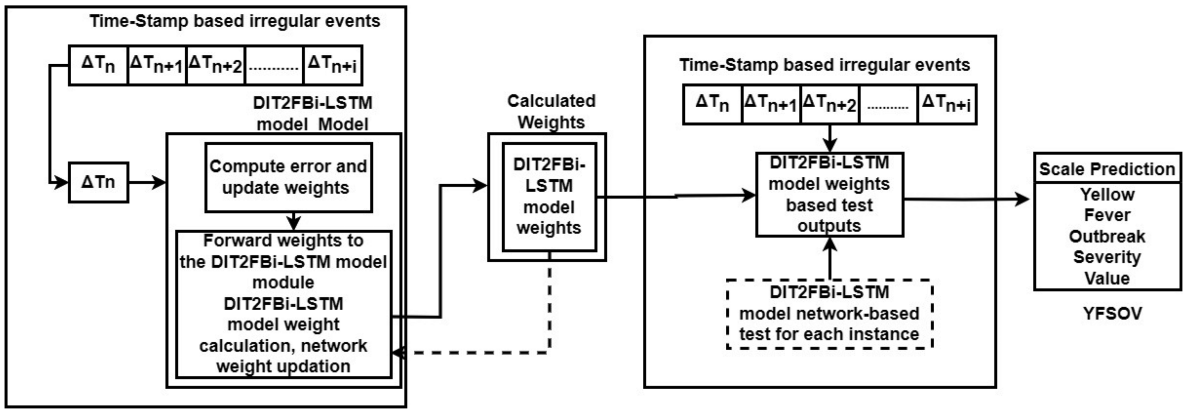

Fig. 5. DIT2FBi-LSTM oriented yellow fever severity prediction.

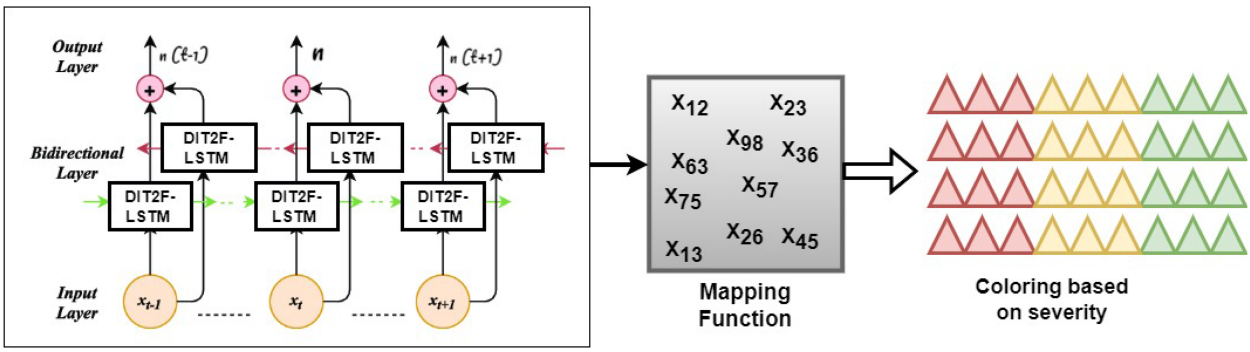

Fig. 6. U-matrix-based SOM visualization and YFOSV value-based color coding.

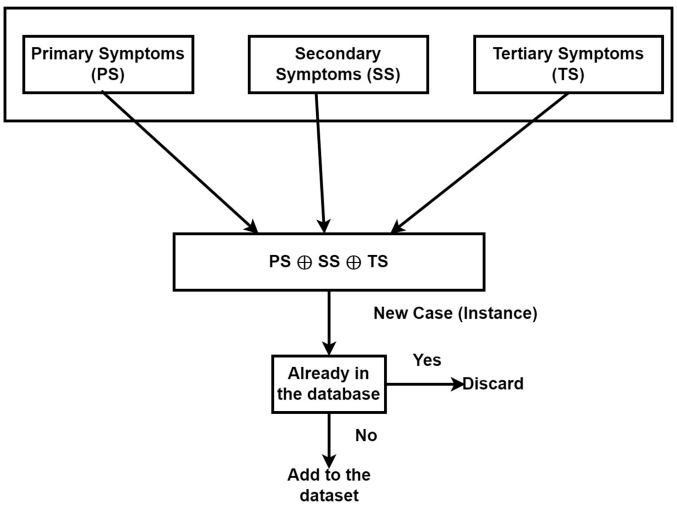

Fig. 7. Data instances formation for yellow fever.

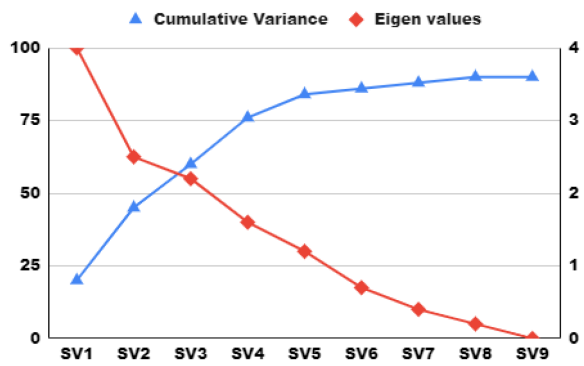

Fig. 8. SVD dimensionality reduction.

demonstrates the abstraction of the UMT method. Generally, the low YFOSV value indicates yellow color, the medium YFOSV value indicates green color, and finally high YFOSV value indicates red color. The steps involved in visualization and alert generation are shown in Algorithm 3.

Timely decision-making in public health services is of utmost importance. Effectively integrating SOM along with GIS provides better interactive visualization. To trace the yellow fever outbreak, enormous location-based sensors can be deployed and GPS coordinates are also captured by mobile devices to draw effective results. Moreover, geospatial information plays a vital role in tracing the yellow fever region so that healthcare authorities, government, and doctors can access and view the information and take preventive actions.

Algorithm 3: Steps for Visualization and Alert Generation.

Input: Recent classification category of user, probability of events, various dataset's attributes, and pre-defined threshold values.

```
The Current category of user, health and location attributes and events for the current time-stamp window.
for i = 1 to n do
  if Current attribute (i) instances > pre-threshold value then
    if category = YF-infected then
      The user is vulnerable and break
    else
      if category == YF-normal then
        Calculate YFSOV value
        if P(YFSOV) > Pre-fixed threshold value then
Generate alert messages to users and nearby hospitals.
        end if
      end if
    end if
  end if
end for
Feed new values to SOM technique for generating the infected region details.
```

III. Performance Evaluation

In this section, the systematic description of various components for generating results is shown below. Moreover, the experiments have been conducted on a system with R studio and the evaluations are executed on Amazon Elastic MapReduce framework, a platform that hosts Hadoop clusters on EC2 memory-optimized i2.xlarge instances provided by Amazon's cluster. The system specifications are Intel Core i7 processor, memory capacity of 16 GB, clock frequency of 2.5 GHz, and 64-bit Windows 11 Pro operating system.

A. Dataset

1) Creation of Yellow Fever Dataset: After an exhaustive search over the Internet for yellow fever datasets, we found 427 records of patients for the year 2017.[2] To conduct experiments and generate effective results for our patient health diagnosis system, we extracted records of individuals. The initial dataset consisted of 122 cases, but we created a simulation environment by bootstrapping these cases to generate more than 2000 cases, as shown in Fig. 7. We used symptom-based datasets to assess the proposed system to ensure that no possible cases were left out. These values can be adjusted based on the simulation at hand if necessary. Moreover, the rigorous approach was also adopted to generate better results at the cloud layer based on the DIT2FBi-LSTM method, discussed in Section III-E.

[2][Online]. Available: http://data.humdata.org/dataset?tags=yellowfever and UCI repository

2) Data Dimensionality Reduction: The best-case datasets generated using Fig. 8 are fed to SVD for dimensionality reduction, using the R Studio tool. The procedure of dimensionality reduction is already shown in Algorithm 1. Fig. 8 shows all nine singular vectors with their eigenvalues and cumulative variance. It depicts that the first four singular vectors have eigenvalues greater the one and cumulatively draw 93.09% variance. These most informative four SVs are forwarded to the fog node for classification and further decision-making.

B. Evaluation Metrics

Metrics, such as specificity, sensitivity, accuracy, and response time, are calculated in each fold and the obtained results are compared with other state-of-the-art algorithms [39].

1) *Accuracy:* The most common and effective performance measure for a classification model is accuracy. It quantifies the proportion of correct predictions among all predictions made. Mathematically, accuracy is expressed as

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{18}$$

where TP (true positive), TN (true negative), FP (false positive), and FN (false negative) represent key performance metrics in the classification process.

2) *Sensitivity:* The sensitivity, also known as TP rate or recall, is the ratio of accurately identified positive instances to the total number of actual positive instances. Mathematically, it is expressed as

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{19}$$

3) *Specificity:* To avoid FP, the proportion of negative instances that were collectively classified as negative determines the specificity

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}}. \tag{20}$$

4) *Average response time:* It is characterized as the mean duration taken by each classification algorithm over ten iterations to produce a diagnosis result.

5) *F-measure:* To maintain the balance between precision and recall, the F-measure is calculated as the harmonic mean between the two. Mathematically, it can be expressed as

$$F - \text{measure} = \frac{2 * \text{Precison} * \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{21}$$

C. k-Fold Cross-Validation Method

To mitigate bias associated with random sampling in training and testing data when comparing the predictive accuracy of multiple methods, a k-fold cross-validation approach was employed. In this methodology, the dataset S is randomly partitioned into k mutually exclusive subsets, denoted as $\{D_1, D_2, \ldots, D_k\}$, each of approximately equal size. The model is trained and tested k times; for each iteration $i \in \{1, 2, 3, \ldots, k\}$, the model is trained on D/D_t and tested on D_t. Accuracy in the cross-validation approach is estimated by calculating the total number of correct

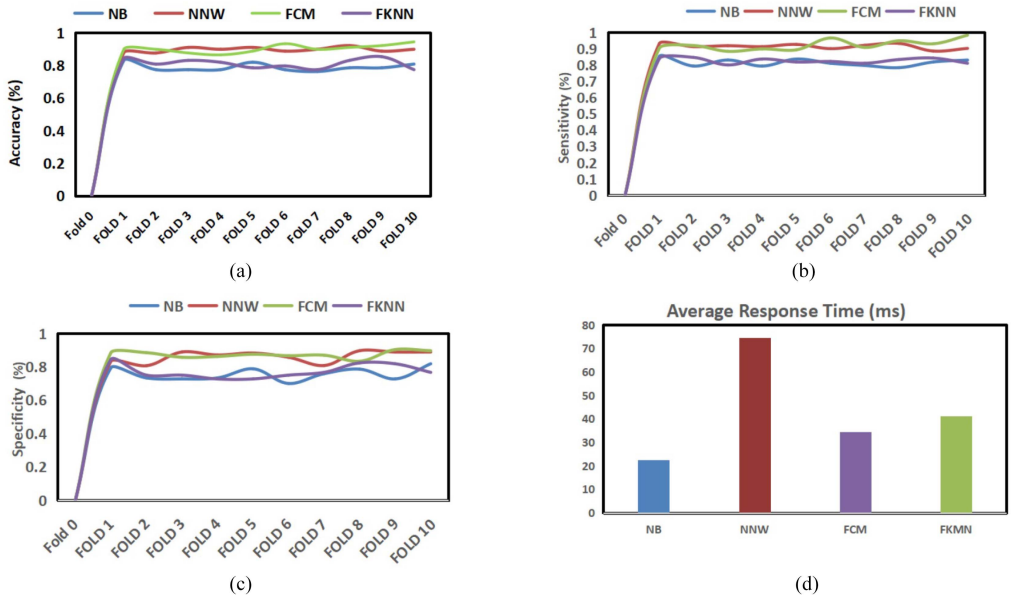

Fig. 9. Statistical results at fog layer: (a) accuracy FCM in comparison with other classifier, (b) sensitivity FCM in comparison with other classifier, (c) specificity FCM in comparison with other classifier, and (d) average response time of all classifiers.

Fold No.	NB				NNW				FCM				FKNN			
	P.matrix	Accuracy	Sensitivity	Specificity	P.matrix	Accuracy	Sensitivity	Specificity	P.matrix	Accuracy	Sensitivity	Specificity	P.matrix	Accuracy	Sensitivity	Specificity
1.	46 7 8 27	0.8295	0.8518	0.7941	54 5 4 25	0.8770	0.9310	0.8330	56 3 6 23	0.8977	0.9032	0.8846	47 5 9 27	0.8409	0.8392	0.8437
2.	46 8 12 22	0.7727	0.7931	0.7333	52 6 5 25	0.8750	0.9122	0.8064	57 4 4 23	0.8977	0.9193	0.8846	44 9 8 27	0.8068	0.8461	0.75
3.	44 9 11 24	0.7727	0.8301	0.7272	56 3 5 24	0.9090	0.9180	0.8888	53 4 7 24	0.875	0.8833	0.8571	46 9 6 27	0.8295	0.8	0.75
4.	46 8 12 22	0.7727	0.7931	0.7333	52 4 5 27	0.8977	0.9122	0.8709	53 4 6 25	0.8636	0.8983	0.8620	46 7 9 26	0.8181	0.8363	0.727
5.	46 7 9 26	0.8181	0.8363	0.7878	50 4 4 30	0.9090	0.9259	0.8823	50 4 6 28	0.8863	0.8928	0.875	45 9 10 24	0.7840	0.8181	0.7272
6.	47 9 11 21	0.7727	0.8103	0.7	54 4 6 24	0.8863	0.9	0.8571	56 4 2 26	0.9318	0.9655	0.8666	46 8 10 24	0.7954	0.8214	0.75
7.	47 9 10 22	0.7613	0.7966	0.7586	58 5 4 21	0.8977	0.9206	0.8076	59 3 6 20	0.8977	0.9076	0.8695	47 7 11 23	0.7727	0.8103	0.7666
8.	47 6 13 22	0.7840	0.7833	0.7857	55 3 4 26	0.9204	0.9322	0.8965	55 5 3 25	0.9090	0.9482	0.8333	45 6 9 28	0.8295	0.8333	0.8235
9.	45 9 10 24	0.7840	0.8181	0.7272	54 3 7 24	0.8863	0.8852	0.8888	53 3 4 28	0.9204	0.9298	0.9032	48 4 9 27	0.8522	0.8421	0.8181
10.	44 8 9 27	0.8068	0.8301	0.8181	55 3 6 24	0.8977	0.9016	0.8888	57 3 2 26	0.9431	0.9827	0.8965	47 7 11 23	0.7727	0.8103	0.7666
Mean		0.7874	0.8142	0.7565		0.8956	0.9138	0.8620		0.9022	0.9230	0.8732		0.8101	0.8257	0.7722
St.SD		0.0228	0.2259	0.0380		0.0146	0.0149	0.0345		0.0246	0.0329	0.0203		0.0283	0.0158	0.0414

Fig. 10. Confusion matrix of Fuzzy-C mean clustering and other state-of-the-art classifiers.

classifications divided by the total number of instances in the dataset. Formally, if $D_{(i)}$ is a test set containing an instance $x_i =< v_i, y_i >$, then the cross-validation accuracy is expressed as follows:

$$\text{acc}_{\text{CV}} = \frac{1}{n} \sum_{(v_i, y_i) \epsilon D} \sigma\left(I(D \backslash D_{(i)}, v_i), y_i\right) \tag{22}$$

where $\sigma(I(D\backslash D_{(i)}, v_i), y_i)$ represents the accuracy measure for each translation, with n denoting the number of translations. Moreover, the accuracy of cross-validation is mainly influenced by the random assignment of individual cases to k different warehouses. A widely adopted strategy to address this involves the stratification of the warehouses themselves [40]. For each subset, a classifier is created using nine out of ten multiples and tested on the tenth to obtain cross-validation of its error rate. The ten cross-validation estimates are then averaged to obtain an estimate of the classifier's accuracy constructed from all data.

In our study, the best 880 cases are considered, and the algorithms are trained and tested using the tenfold cross-validation method. The rating is based on the accuracy measures discussed above (accuracy, sensitivity, and specification). The results are based on the average results obtained from the test dataset (tenthfold) for each fold, as shown in Fig. 9. Naive Bayes (NB), nearest neighbor with weights (NNW), Fuzzy-C means clustering, and Fuzzy-k nearest neighbor (FKNN) are used to validate our model. Comparing these models, we found that NB achieved a classification accuracy of 78.74% with a sensitivity and specificity of 80.82% and 75.89%, respectively. The NNW model achieved a classification accuracy of 89.56% with a

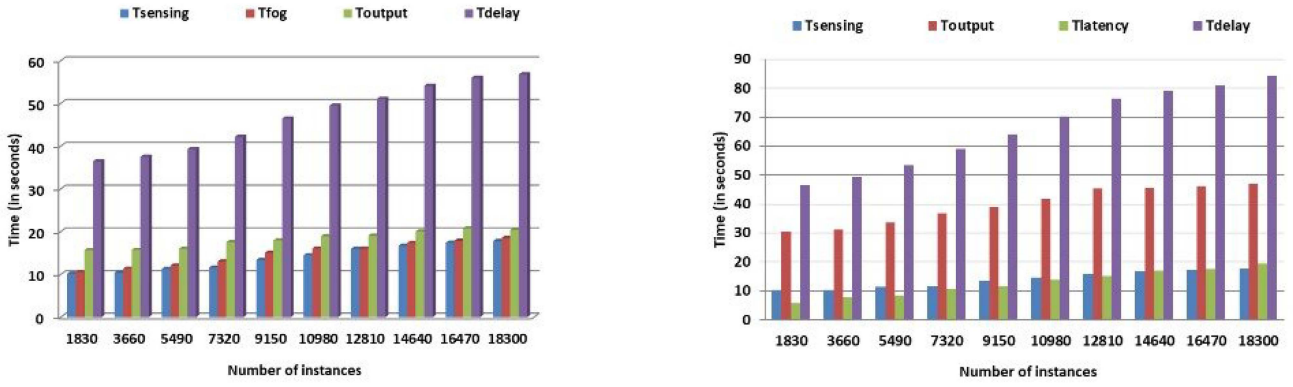

Fig. 11. Temporal delays for (a) the proposed model and (b) the model without fog layer.

sensitivity and specificity of 91.86% and 80.87%, respectively, whereas the FKNN model achieved a classification accuracy of 81.01% with a sensitivity and specificity 82.57% and 77.22%, respectively. Moreover, the confusion matrix for the above-mentioned state-of-the-art classifiers in Fig. 10 also reveals the efficacy of the FCM in the proposed methodology.

D. Temporal Efficiency

Temporal efficiency holds significance in applications where real-time decisions are crucial for delivering time-sensitive information to end users. It measures the overall time taken by the system to generate the required output, encompassing various components: data sensing (T_{sensing}), data preprocessing and classification (T_{fog}), and data analysis and decision making (T_{output}). The total delay time (T_{delay}) is mathematically expressed as the sum of these factors

$$T_{\text{delay}} = T_{\text{sensing}} + T_{\text{fog}} + T_{\text{output}}. \tag{23}$$

In Fig. 11(a), the results of the proposed system are illustrated, with a focus on time delay in the presence of a fog layer. The x-axis emphasizes the increase in data instances and their impact on the total delay, whereas the y-axis denotes the time for each component in seconds. The results reveal the average time (T_{delay}) of 46.94 s throughout the entire experimental process, up to 18 300 instances. Moreover, further analysis shows that the data sensing component averaged approximately 13.93 s, followed by data preprocessing and classification consuming an average of 14.79 s. Finally, the data analysis and decision-making component required an average of 18.22 s to generate the results.

In the analysis of the temporal efficiency of our proposed system, significant modifications were introduced. Specifically, the responsibility for data preprocessing and classification at the fog layer was shifted to the cloud layer, which now also encompasses predictive analysis. As depicted in Fig. 11(b), this alteration illustrates the delay in data access and the final prediction in the proposed system. A latency of an average of 12.67 s is observed due to the data flow from the data acquisition layer to the cloud layer. Consequently, the overall average time is noted to be around 66.318 s. This evaluation underscores the pivotal role of the fog layer in addressing real-time data processing and latency concerns.

TABLE III
DIT2Bi-LSTM HYPERPARAMETER WITH THEIR OPTIMAL VALUES

Hyperparameters	Values (%)
Number of epochs	100
Learning rate	0.004
Dropout rate	0.02
Batch size	32
Filter size	3*3
Pooling size	2*2
DIT2Bi-LSTM units	128

TABLE IV
COMPARATIVE ANALYSIS OF PROPOSED PREDICTIVE MODEL WITH CONVENTIONAL METHODS

Method	AUC (%)	CI (%)	Recall	Precision	F–1
IT2FLS	80.07	[74–79]	82.44%	83.31%	83.57%
LSTM	84.73	[82–86]	87.32%	88.07%	88.23%
Fuzzy-LSTM	86.42	[84–89]	88.55%	89.92%	89.11%
IT2ANFIS	87.21	[87–91]	89.27%	89.84%	89.54%
DIT2FLSTM	90.39	[88–92]	91.63%	90.78%	90.03%
DIT2FBi-LSTM Model	93.05	[91–96]	93.17%	92.43%	92.53%

E. Statistical Evaluation of DIT2Bi-LSTM Model

This section focuses on the time-series data PM (DIT2Bi-LSTM). The parameter tuning for the proposed predictive model is given in Table III. Applying these parameters, a comparative analysis is carried out with other conventional methods. Table IV demonstrates the performance comparison based on statistical parameters with counterpart models, which include IT2FLS, LSTM, Fuzzy-LSTM, IT2ANFIS [41], and DIT2FLSTM for the yellow fever time-series prediction. Moreover, a two-sample t-$test$ has also been considered for the applicability of the proposed predictive model. The null hypothesis [42], [43] was defined as $P_0 = \mu_i > \mu_j$ and $P_1 = \mu_i < \mu_j$, where μ_i and μ_j are the means of the area under the ROC curve (AUC) of the proposed predictive model ($\mu_i = 1/10 \sum_{k=1}^{10} \text{AUC}_j$) and IT2FLS as per ten-cross fold validation assessment. The obtained results

TABLE V
PARAMETER-BASED COMPARATIVE ANALYSIS OF DEEP LEARNING MODELS

Parameter	Convolutional neural networks (CNNs)	Long short-term memory (LSTM)	Recurrent neural networks (RNNs)
Definition	A type of neural network optimized for processing grid-like data such, as images, using convolutional layers to detect features and patterns.	An advanced RNN capable of learning long-term dependencies in sequence data through specialized gating mechanisms.	A neural network designed to handle sequential data, with connections that loop back, allowing it to use previous information to inform current processing.
Memory	No internal memory.	Long-term memory.	Short-term memory.
Overfitting	Less prone due to weight sharing.	More prone due to complex architecture and longer training times.	More prone due to sequential dependencies.
Advantages	Excellent for extracting spatial features, translational invariance, parameter sharing, and robust to input variation.	Effective in capturing long-range dependencies, mitigates vanishing gradient problems and is suitable for sequential tasks with complex patterns.	Simple and efficient for processing sequential data, suitable for real-time applications, and can handle varying lengths of input sequences.
Applications	Image recognition, object detection, image segmentation, natural language processing.	Speech recognition, language translation, text generation, time-series prediction.	Language modeling, handwriting recognition, video analysis, sequential data classification.
Complexity	Simple layers	Complex with gates, but better at capturing long-term dependencies.	Simple but recurrent connections.
Internal Structure Components	Input layer, conventional layer, Activation Layer, Pooling Layer, Fully connected Layer, and output layer.	Input layer, LSTM Cell consists of four components (Input modulation layer, Input gate, Forget gate, and Output gate), and Output layer.	Input layer, Recurrent Layer, Activation Layer, Output layer.

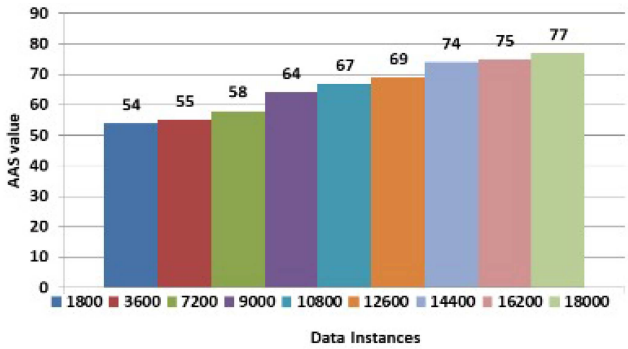

Fig. 12. System stability with varying data instances.

show the efficacy of the predictive model with a mean of 0.97 (IT2FLS mean is 0.82) and the t-test failed to reject the null hypothesis.

F. Stability Analysis

Stability analysis is a crucial parameter for assessing the consistency of the proposed system over its operational duration. The stability value, ranging from 0 to 100, is indicative of the system's performance reliability. In the yellow fever environment, a healthy average absolute shift (ASS) concerning the increase in the number of data instances serves as a key metric. As illustrated in Fig. 12, the system consistently maintains an ASS of greater than 70 throughout its lifespan or evaluation period. This observation underscores the robustness and reliability of the proposed system in the context of a yellow fever environment. The stability results presented in Fig. 12 affirm the efficacy and suitability of the proposed system for addressing challenges related to yellow fever.

G. Discussions

Experimental results of SVD and FCM at the fog layer and predictive model at the cloud layer revealed the efficacy of the proposed yellow fever prediction framework. SVD's first four singular vectors have been utilized for better results at the fog layer. FCM classifier outperforms other state-of-the-art models in terms of accuracy and other statistical parameters results. The latency issues without the fog layer were also considered for better decision-making at the cloud layer. Moreover, the predictive model is showing better statistical results compared with other counterparts: 13.5% greater than IT2FLS, 9% greater than LSTM, 7.5% greater than fuzzy-LSTM, 7% greater than IT2ANFIS, and around 2.5% greater than DIT2FLSTM in terms of AUC. The predictive model being proposed demonstrates an average AUC of 93.6%, with a 95% confidence interval ranging from 91% to 96%, in forecasting yellow fever outbreak occurrences.

To gain deeper insights into the application domains and internal mechanisms of deep learning models, such as RNNs, CNNs, and LSTM, a comprehensive comparative analysis was conducted, as detailed in Table V. Moreover, the envisioned horizon for the proposed predictive model suggests its potential applicability in biomedical fields, encompassing healthcare and biopharmaceutical research, particularly within the realm of time-series analysis.

IV. CONCLUSION AND FUTURE WORK

With the progression of medical devices utilizing IoT-FC, data analytics, and CC, numerous interactions can be efficiently and effectively managed. This article focuses on incorporating novel methodology to handle yellow fever disease. SVD and FCM classifier play a pivotal role at the fog layer in generating yellow fever cases. Moreover, a novel DIT2FBi-LSTM predictive model is proposed to deal with uncertainties in time-series data for better decision-making at the cloud layer. In addition to that, the SOM-based UMT, alert-based processing, and utilizing the GPS for precise prediction and severity calculation also increase the efficacy of the proposed system. The proposed predictive model gets an accuracy of more than 93% for large samples and can be easily updated with new cases because of its deep architecture. Further studies can be carried out in the future to tune the predictive model cell structure for other real-world applications with nonstationary certainties. The evolutionary algorithms or optimization techniques can be incorporated in the future for the better performance of the proposed predictive model.

REFERENCES

[1] L. d. G. d. Lataillade et al., "Risk of yellow fever virus transmission in the Asia-Pacific region," *Nature Commun.*, vol. 11, no. 1, 2020, Art. no. 5801.

[2] S. Vanderslott and T. Marks, "Travel restrictions as a disease control measure: Lessons from yellow fever," *Glob. Public Health*, vol. 16, no. 3, pp. 340–353, 2021.

[3] E. Gianchecchi, V. Cianchi, A. Torelli, and E. Montomoli, "Yellow fever: Origin, epidemiology, preventive strategies and future prospects," *Vaccines*, vol. 10, no. 3, 2022, Art. no. 372.

[4] P. Verma, A. Gupta, M. Kumar, and S. S. Gill, "FCMCPS-COVID: AI propelled fog–cloud inspired scalable medical cyber-physical system, specific to Coronavirus disease," *Internet Things*, vol. 23, 2023, Art. no. 100828.

[5] L. Hussain-Alkhateeb et al., "Early warning systems (EWSS) for chikungunya, dengue, malaria, yellow fever, and Zika outbreaks: What is the evidence? A scoping review," *PLoS Neglected Trop. Dis.*, vol. 15, no. 9, 2021, Art. no. e0009686.

[6] N. K. Mehto, P. Sharma, S. Kumar, M. Khanuja, R. Rawal, and J. Narang, "Towards papertronics based electrode decorated with zinc oxide nanoparticles for the detection of the yellow fever virus consensus sequence," *Process Biochem.*, vol. 123, pp. 36–43, 2022.

[7] P. Verma and S. K. Sood, "Fog assisted-IoT enabled patient health monitoring in smart homes," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1789–1796, Jun. 2018.

[8] N. Kaur, S. K. Sood, and P. Verma, "Cloud resource management using 3 Vs of Internet of Big Data streams," *Computing*, vol. 102, pp. 1463–1485, 2020.

[9] A. Hazra, P. Rana, M. Adhikari, and T. Amgoth, "Fog computing for next-generation Internet of Things: Fundamental, state-of-the-art and research challenges," *Comput. Sci. Rev.*, vol. 48, 2023, Art. no. 100549.

[10] A. R. Nandhakumar et al., "EdgeAISim: A toolkit for simulation and modelling of AI models in edge computing environments," *Meas.: Sensors*, vol. 31, 2024, Art. no. 100939.

[11] R. Singh et al., "Edge AI: A survey," *Internet Things Cyber- Phys. Syst.*, vol. 3, pp. 71–92, 2023.

[12] A. Chakraborty, M. Kumar, N. Chaurasia, and S. S. Gill, "Journey from cloud of things to fog of things: Survey, new trends, and research directions," *Softw.: Pract. Experience*, vol. 53, no. 2, pp. 496–551, 2023.

[13] J. Singh, P. Singh, and S. S. Gill, "Fog computing: A taxonomy, systematic review, current trends and research challenges," *J. Parallel Distrib. Comput.*, vol. 157, pp. 56–85, 2021.

[14] S. S. Gill et al., "AI for next generation computing: Emerging trends and future directions," *Internet Things*, vol. 19, 2022, Art. no. 100514.

[15] S. K. Sood and I. Mahajan, "Fog-cloud based cyber-physical system for distinguishing, detecting and preventing mosquito borne diseases," *Future Gener. Comput. Syst.*, vol. 88, pp. 764–775, 2018.

[16] V. Vijayakumar, D. Malathi, V. Subramaniyaswamy, P. Saravanan, and R. Logesh, "Fog computing-based intelligent healthcare system for the detection and prevention of mosquito-borne diseases," *Comput. Hum. Behav.*, vol. 100, pp. 275–285, 2019.

[17] S. Manoharan, K. M. Kumar, and N. Vadivelan, "A novel CNN-TLSTM approach for dengue disease identification and prevention using IoT-fog cloud architecture," *Neural Process. Lett.*, vol. 55, no. 2, pp. 1951–1973, 2023.

[18] P. D. Singh, R. Kaur, K. D. Singh, G. Dhiman, and M. Soni, "Fog-centric IoT based smart healthcare support service for monitoring and controlling an epidemic of swine FLU virus," *Informat. Med. Unlocked*, vol. 26, 2021, Art. no. 100636.

[19] J. L. Sarkar et al., "FogCom: SDN-enabled fog node selection for early detection of communicable diseases," *J. King Saud Univ.- Comput. Inf. Sci.*, vol. 35, no. 6, 2023, Art. no. 101432.

[20] T. A. Ahanger, U. Tariq, M. Nusir, A. Aldaej, I. Ullah, and A. Sulman, "A novel IoT-fog-cloud-based healthcare system for monitoring and predicting COVID-19 outspread," *J. Supercomput.*, vol. 78, no. 2, pp. 1783–1806, 2022.

[21] B. U. Demirel, I. A. Bayoumy, and M. A. Al Faruque, "Energy-efficient real-time heart monitoring on edge-fog-cloud Internet of Medical Things," *IEEE Internet Things J.*, vol. 9, no. 14, pp. 12472–12481, Jul. 2022.

[22] M. Kumar, A. Rai, Surbhit, and N. Kumar, "Autonomic edge cloud assisted framework for heart disease prediction using RF-LRG algorithm," *Multimedia Tools Appl.*, vol. 83, pp. 5929–5953, 2023.

[23] I. Nelson, C. Annadurai, and K. N. Devi, "An efficient AlexNet deep learning architecture for automatic diagnosis of cardio-vascular diseases in healthcare system," *Wireless Pers. Commun.*, vol. 126, no. 1, pp. 493–509, 2022.

[24] A. Pravin, T. P. Jacob, and G. Nagarajan, "An intelligent and secure healthcare framework for the prediction and prevention of dengue virus outbreak using fog computing," *Health Technol.*, vol. 10, pp. 303–311, 2020.

[25] C. Jatoth, E. Neelima, A. Mayuri, and S. R. Annaluri, "Effective monitoring and prediction of Parkinson disease in smart cities using intelligent health care system," *Microprocessors Microsyst.*, vol. 92, 2022, Art. no. 104547.

[26] B. Costa, J. BachiegaJr., L. R. de Carvalho, and A. P. Araujo, "Orchestration in fog computing: A comprehensive survey," *ACM Comput. Surv.*, vol. 55, no. 2, pp. 1–34, 2022.

[27] C. Yang and Q. Shi, "An interval perturbation method for singular value decomposition (SVD) with unknown-but-bounded (UBB) parameters," *J. Comput. Appl. Math.*, vol. 436, 2024, Art. no. 115436.

[28] J. Li, Q. Zhou, Y. He, H. Williams, and H. Xu, "Driver-identified supervisory control system of hybrid electric vehicles based on spectrum-guided fuzzy feature extraction," *IEEE Trans. Fuzzy Syst.*, vol. 28, no. 11, pp. 2691–2701, Nov. 2020.

[29] B. Sindhusaranya and M. R. Geetha, "Retinal blood vessel segmentation using root guided decision tree assisted enhanced Fuzzy-C mean clustering for disease identification," *Biomed. Signal Process. Control*, vol. 82, 2023, Art. no. 104525.

[30] F. Di Martino and F. Delmastro, "Explainable AI for clinical and remote health applications: A survey on tabular and time series data," *Artif. Intell. Rev.*, vol. 56, no. 6, pp. 5261–5315, 2023.

[31] D. Wu, "On the fundamental differences between interval type-2 and type-1 fuzzy logic controllers," *IEEE Trans. Fuzzy Syst.*, vol. 20, no. 5, pp. 832–848, Oct. 2012.

[32] J. M. Mendel, R. I. John, and F. Liu, "Interval type-2 fuzzy logic systems made simple," *IEEE Trans. Fuzzy Syst.*, vol. 14, no. 6, pp. 808–821, Dec. 2006.

[33] D. Wu and J. M. Mendel, "Recommendations on designing practical interval type-2 fuzzy systems," *Eng. Appl. Artif. Intell.*, vol. 85, pp. 182–193, 2019.

[34] N. Sharma, R. K. Sunkaria, and A. Kaur, "Electrocardiogram heartbeat classification using machine learning and ensemble convolutional neural network-bidirectional long short-term memory technique," *IEEE Trans. on Artif. Intell.*, early access, Oct. 16, 2023, doi: 10.1109/TAI.2023.3324627.

[35] A. Andreas, C. X. Mavromoustakis, H. Song, and J. M. Batalla, "Optimisation of CNN through transferable online knowledge for stress and sentiment classification," *IEEE Trans. Consum. Electron.*, vol. 70, no. 1, pp. 3088–3097, Feb. 2024.

[36] A. Jaiswal and R. Kumar, "Breast cancer diagnosis using stochastic self-organizing map and enlarge C4.5," *Multimedia Tools Appl.*, vol. 82, no. 12, pp. 18059–18076, 2023.

[37] O. J. Akindote, A. O. Adegbite, S. O. Dawodu, A. Omotosho, A. Anyanwu, and C. P. Maduka, "Comparative review of Big Data analytics and GIS in healthcare decision-making," *World J. Adv. Res. Rev.*, vol. 20, no. 3, pp. 1293–1302, 2023.

[38] D. Nagar, P. Ramu, and K. Deb, "Visualization and analysis of pareto-optimal fronts using interpretable self-organizing map (ISOM)," *Swarm Evol. Comput.*, vol. 76, 2023, Art. no. 101202.

[39] M. Kumar, G. K. Walia, H. Shingare, S. Singh, and S. S. Gill, "AI-based sustainable and intelligent offloading framework for IIoT in collaborative cloud-fog environments," *IEEE Trans. Consum. Electron.*, vol. 70, no. 1, pp. 1414–1422, Feb. 2024.

[40] X. Zhang and C.-A. Liu, "Model averaging prediction by k-fold cross-validation," *J. Econometrics*, vol. 235, no. 1, pp. 280–301, 2023.

[41] A. Safari, M. Mazinani, and R. Hosseini, "A novel type-2 adaptive neuro fuzzy inference system classifier for modelling uncertainty in prediction of air pollution disaster (research note)," *Int. J. Eng.*, vol. 30, no. 11, pp. 1746–1751, 2017.

[42] R. Hosseini, S. D. Qanadli, S. Barman, M. Mazinani, T. Ellis, and J. Dehmeshki, "An automatic approach for learning and tuning Gaussian interval type-2 fuzzy membership functions applied to lung CAD classification system," *IEEE Trans. Fuzzy Syst.*, vol. 20, no. 2, pp. 224–234, Apr. 2012.

[43] A. Safari, R. Hosseini, and M. Mazinani, "A type-2 fuzzy time series model for pattern similarity analysis: A case study on air quality forecasting," *IEEE Intell. Syst.*, vol. 37, no. 2, pp. 92–102, Mar./Apr. 2022.