International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Machine Learning and Cloud Computing Based Adaptable Structure for Intelligent Covid Monitoring in the Work Environment

[1]Rajendra Pandey P, [2]Sachin Jain, [3]Mehvish Khalid, [4]Suchithra R

Submitted: 22/04/2023 **Revised**: 19/06/2023 **Accepted**: 27/06/2023

Abstract: As a result of the new coronavirus outbreak's global expansion and the respiratory diseases it causes in people, COVID-19 has become a major global pandemic. The only way to stop this spread, according to the World Health Organization, is to increase testing and isolate those who are sick. In the meantime, the clinical testing that is now being used is time-consuming. Systems for remote diagnosis may be useful in this situation. The healthcare industry generates a large quantity of data, which we process using certain machine learning algorithms to identify the presence of illness. Several IoT-enabled sensors are accessible to detect the patient's entire information about a specific person's behavior, human anatomy, and physiology. The information gathered by the sensors is sent to the internet and linked to a cloud server. Physicians may access patient records stored on the web server and preserve them there, giving them access to the information from anywhere on the globe. Any unexpected change in a patient's data, while they are using the healthcare system, will unavoidably result in the patient's data being immediately uploaded to the appropriate doctor. In rural and distant places, this kind of healthcare system would be most beneficial. We proposed a novel butterfly-optimized multitemporal support vector machine (BO-MTSVM) approach to overcome the aforementioned problems. The suggested technique performs better than other current methods in COVID monitoring, according to simulation data.

Keywords: *COVID-19, Internet of Things (IoT), covid monitoring system, big data, butterfly-optimized multitemporal support vector machine (BO-MTSVM)*

1. Introduction

People's daily life was significantly impacted by the COVID-19 pandemic, enabling it challenging for them to access medical care, get a job, or go to school. A global epidemic is about to finish now that the outbreak has changed into a major worldwide health concern. But the public is still at risk from COVID-19's widespread signs, which include coughs, headaches, eye discomfort, dizziness, exhaustion, and death. Government officials and health organizations around the world placed a major emphasis on health measures and prevention strategies to stop the deadly virus's rapid expansion. Inadequate access to basic amenities, slums, and destitute communities, as well as the volume and duration of public transit, as well as the state of a nation's infrastructure, are all factors that could make COVID-19 instances more prevalent, especially from a sociodemographic perspective [1]. It can classify and predict patients based on their susceptibility or resistance to potential infections by using machine-learning inspections of genetic variations from COVID-19 cases. Remote patient monitoring and reliable tracking of vital signs, blood sugar levels, and weight changes in clinical trials are two ways that the Internet of Medical Things (IoMT) improves patient-centric initiatives in clinical settings [2]. However, an infection can potentially result in severe pneumonia, and in that case, the patient might not survive. This makes it necessary to note that the coronavirus sickness can progress without producing any signs on the body of the patient. Infected with a throat for a few days, this virus spreads to the patient's lungs, where it may have devastating consequences and cause fever, coughing, and excruciating headaches [3]. It's critical to remember that the deployment of intelligent COVID monitoring systems must adhere to all applicable privacy and data protection laws [4]. Wearable sensors, including temperature sensors, may be utilized to detect the coronavirus disease, as may hospital-based diagnostic procedures. However, it is typically challenging to identify useful risk indicators for covid-19 from temperature sensor data using some APIs (Application Programming Interfaces) or digital health testing [5]. To establish confidence and adherence to these frameworks in the workplace, open communication and transparency with employees regarding the goal, usage, and security of the obtained data are essential. The main goals of intelligent

[1]*Assistant Professor, College of Computing Science and Information Technology, Teerthanker Mahaveer University, Moradabad, Uttar Pradesh, India, Email id: panday_004@yahoo.co.uk*
[2]*Assistant professor, School of Computer Science & System, JAIPUR NAITONAL UNIVERSITY, JAIPUR, India, Email Id: sachin.jain@jnujaipur.ac.in*
[3]*Professor, School of Nursing & Paramedical Sciences, Dev Bhoomi Uttarakhand University, Uttarakhand, India, Email Id: dean.sonps@dbuu.ac.in*
[4]*Professor, Department of Computer Science and IT, Jain(Deemed-to-be University), Bangalore-27, India, Email Id: r.suchithra@jainuniversity.ac.in*

COVID monitoring in the workplace are to protect staff members' health and well-being while lowering the chance of COVID-19 transmission.

The further part of the study includes phase 2 indicates the suggested technique, phase 3 indicates the relatedworls, phase 4 indicates the result and discussion and phase 5 indicates the conclusion.

2. Related work

The study [6] created and executed a sophisticated health monitoring and diagnostic method for COVID-19 patients with serious heart arrhythmias. The research [7] suggested an intelligent healthcare system with cloud and IoT integration. In this concept, deep learning (DL) and smart connection sensors are combined to enable smart decision-making from the viewpoint. The study [8] introduced AI and IoT technology which exist crucial in minimizing the negative consequences of this sickness and promoting its recovery. Before learning about new technology and potential remedies, it examines the economic effects of COVID-19. With the aid of cloud services, mobile apps, an intelligent IoT structure, AI, machine learning, and 5G technology, the research's main suggesting is a solution for COVID-19 self-analysis, self-evaluating, and self-control via personal mobiles and personal data. The study [9] investigated an "EdgeSDN-I4COVID" design for the COVID-19 management of the smart industry while taking into account IoT networks. The paper also discusses how to effectively and automatically track IoT data from a distance using SDN-enabled layers including data, control, and applications. The research [10] suggested a BPMN expansion to allow IoT-aware business process (BP) modeling. By utilizing the intelligent covid monitoring systems, a suggested BPMN 2.0 addition. Then, to conduct a series of in-depth tests that demonstrate the effectiveness and efficiency of our research, they offer prototypes for these two smart systems. The study [11] suggested using a wireless smart home healthcare support system (ShHeS) to track patients' health and fill medications while they are at home. The research [12] presented a potential IoT use case in healthcare and physical distance tracking for virus conditions. The recommended method includes fog-dependent on Machine Learning (ML) devices for information assessment and diagnostics, lightweight & low-cost IoT nodes, and a smartphone app. The study [13] proposed an IoT-based system that employs these recorded values to be a real-time health monitoring system. The device's LCD shows the current temperatures, heart rate, and oxygen saturation levels and is easily connected to a mobile app for quick access. The research [14] offered a method for actual disease identification and observation. The proposed method would gather real-time signs data from users using an IoT structure to manage the therapy responses of those that remain virus-free after already recovering, to recognize distrusted covid19 cases earlier, and to comprehend the virus by gathering and examining relevant information. The study [15] discussed healthcare delivery during the COVID-19 epidemic may be complemented and supported by robotics. The research [16] suggested a hybrid framework combining artificial intelligence (AI), humanoid robots, and the Internet of Things (IoT) to create a smart and secure working environment. This structure includes a list of functional hierarchy modules along with rules and principles which offer a structural description of the accumulation in the workplace. Also underlined for the structure setup are ethical issues.

3. Methodology

An adaptive structure for intelligent COVID monitoring in the workplace often refers to a framework or system that uses a variety of tools and techniques to keep track of and respond to COVID-19 incidents and hazards in the workplace. A structure like this attempts to improve security, slow the spread of the virus, and guarantee employee welfare.

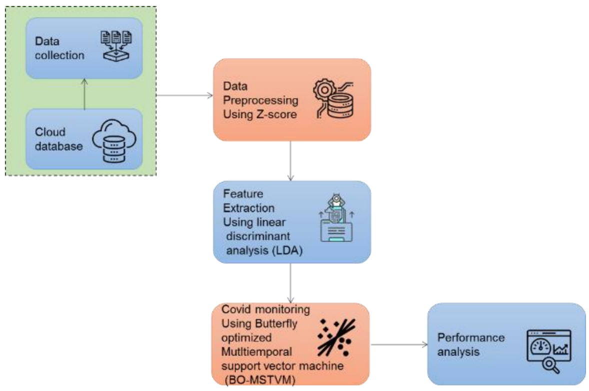

Fig.1. Representation of the proposed methodology

Figure 1 denotes the depiction of the proposed methodology. In this section, we used to collect the data from a cloud database and preprocessing technique takes place, and the BO-MTSVM approach is implemented in detail.

3.1 Dataset collection

We collected the data from cloud storage. We utilized a made-up dataset to diagnose COVID-19 patients in a hospital. 399 patients with a validated COVID-19 infection who were hospitalized at the Pondicherry Institute of Medical Sciences were added to the data set that was provided. Vital signs, employee data, chronic disease, indicators, and other information are among the attributes in this dataset. The development of the patient's health will be tracked using the data provided.

3.2 Preprocessing using z-score

Z-score normalization uses the data's mean and standard deviation as its starting points. In cases when the lowest and maximum values of the data are unknown, this technique is of great use. This is the formula that is used:

$$Y_{new} = \frac{y-\mu}{\sigma} = \frac{y-Mean(Y)}{stdDev(Y)} \tag{1}$$

Y_{new} =The adjusted value obtained after scaling the data

Y = outdated value

μ = Statistics mean

σ = Estimated Standard Deviation

3.3 Feature extraction using linear discriminant analysis (LDA)

Especially for classification applications, linear discriminant analysis (LDA) may also be utilized as a feature extraction method. The goal of feature extraction utilizing LDA in this situation, as opposed to topic modeling, is to identify lower-dimensional representations which improve class separation. To decrease the problem's dimensionality and increase the pattern classifier's capacity to generalize while at the same time lowering the computing processing demands, we apply a linear discriminant analysis. This step of feature extraction may be described as a transformation matrix T^6-based mapping from a d-dimensional input space (Q^c) to an m-dimensional feature space (Q^n).

$$S: Q^c \to Q^n \; n < c \tag{2}$$

By linearly translating the input space to the output space, the linear discriminant analysis achieves feature extraction. The Fisher linear discriminant classifier (FLD) and nearest mean classifier (NMC), often known as the Euclidean distance classifier, are the often well-known and frequently employed linear discriminant classifiers. Both classifiers' creation processes are essentially identical, with a few tiny changes towards the end.

$(V_j)_I$ Represents a vector that represents the ith trained samples produced by the class j discrete wavelet transform. The value of i will either be 1 or 2, where j = 1 denotes incident-free samples and j = 2 denotes incident samples, in the traffic incident-detection instance. Also, i = 1, 2,..., m_i, where m_j is the number of practice samples in group j, and $m = m_1 + m_2$, is the total quantity of training samples. The following is a definition of the dimensions d within-class covariance square matrix:

$$(V_j)_I = (v_{j,i}^1, V_{j,i}^s, \ldots\ldots, v_{j,i}^6)$$

$$j = 1,2, \ldots\ldots m_i, m = m_1 + m_2 \tag{3}$$

Where m_j class j is the mean vector. Data must be classified into areas with incidents and regions without incidents to solve the incident recognition issue, which is a two-class issue. The dimension d between-class covariance square matrix D_u is described as follows for this two-class issue:

$$D_u = \frac{1}{m}\sum_{i=2}^{2}\sum_{j=1}^{m_i}\left[(V_j)_I - n_i\right]\left[(V_j)_I - n_i\right]^S \tag{4}$$

Here m represents the average vector for all information. Finding a transformation matrix T with the within-class scatter reduced and the among-class scatter increasing is the aim of linear discriminant analysis.

$$D_A = \frac{m_1}{m}(n_1 - n)(n_1 - n)^S + \frac{m_2}{m}(n_2 - n)(n_2 - n)^S$$

$$D_A = \frac{m_1 m_2}{m^2}(n_2 - n_1)(n_2 - n_1)^S \tag{5}$$

Since D_A is a function of only one vector ($n_2 - n_1$), its rank is one. And since D_u has a full rank, its inverse exists, and the rank o$D_U^{-1}D_A$ is also equal to one. The only nonzero eigenvalue it possesses is one. For this nonzero eigenvalue, the associated eigenvector is

$$F_1 = \frac{D_u^{-1}(n_2 - n_1)}{\|D_u^{-1}(n_2 - n_1)\|} \tag{6}$$

Wherein the eigenvector's norm is made to have a single value by selecting a constant denominator; the mapping function which yields the output vector Z for our two-class incident-detection problem is an eigenvector that only works on one vector and necessitates a single discriminating feature.

$$Z = F_J^S V = d(n_2 - n_1)^S D_u^{-1} V \tag{7}$$

Here, d is a constant.

3.4 Butterfly-optimized multitemporal support vector machine (BO-MTSVM) approach

The butterfly optimization algorithm (BOA) is based on a fusion of the biological characteristics and behaviors of the butterfly insect. Lepidoptera is a class of insects that includes butterflies. It is capable of hearing, touching, tasting, and seeing. It imitates friendly interactions and fatty foods. These senses enable it to find food, find a mate, flee from danger, and move from one location to another. It produces an intense smell while immigrating to get farther. The second-best butterfly is drawn to the fragrance's intensity by detecting it and traveling toward it; this process is known as a global search. Local search is when a search engine moves to a different new spot in the search space if it is unable to detect the strength of the aroma of the best butterfly at random. The physical capacity of the stimulus is demonstrated by aroma sensing, which can be stated as:

$$oeq_j = tJ^b \tag{8}$$

Here, oeq_j denotes the perceived fragrance intensity, s is the parametric value for sense, J is the stimulus fragrance intensity, b is the power exponent of fragrance absorption, and Modality of sensor t. The three steps of the butterfly movements are the evaluation of the answer, the global search, and the local search. In the stage of the global search, depending on the strength of the smell, it can draw in another butterfly and move to choose the best butterfly, and it may be stated as follows:

$$z_j^{s+!} = z_j^s + \left(qmc^2 a^* - z_j^s\right) eq_j \tag{9}$$

Where, z_j^s denotes the value of the vector, at each repetition, it indicates the solution (a butterfly), s, a* is the overall best answer, rnd is a random number between [0, 1], and eq_j denotes the fragrance intensity of jth butterfly. The minimum fitness value is established in this global search phase a, which is in all result answers and adjusts its velocity by fitness value. The following terms are explained in the local search phase:

$$z_j^{s+!} = z_j^s + \left(qmc^2 z_i^s - z_j^s\right) eq_j \tag{10}$$

z_i^s and z_l^s are two vector values that reflect the different species of butterflies found in identical populations. Its parameters should be updated for the best butterfly as follows:

$$b(it) = b_{is} - \left(b_{jt} - b_{ej}\right) \times sin\left(\frac{\pi}{\mu} \times \left(\frac{it}{njsq}\right)^2\right) \tag{11}$$

Where b_{jt} and b_{ej} denote the parametric initial and final values. The tuning parameter values are b and l, and the maximum amount of iterations is mitr. By the fitness value, adjust its position. Following definitions pertain to the local search phase:

$$z_s^{s+1} = z_j^s + wf_j^{s+1} \tag{12}$$

Where z_i^s and z_l^s denote the velocity of the ith element at iteration s and s+1. Algorithm 1 describes the pseudocode method. Algorithm 1 initializes parametric variables and chooses the optimal option depending on the fragrance butterfly's intensity value.

Algorithm 1: Butterfly Optimization Algorithm

Input: Create a baseline for the butterfly populations m, the power exponent of scent is the parametric parameter b, absorption, maximum iteration$s$, and modality of sensor t.

Output: Optimal Solution

Begin the counter variable $it = 0$

For $(j - j < m)$ do

Create the first butterfly population at randomly z_j^s.

The fitness function should be calculated for every butterfly $e\left(z_j^s\right)$

Utilizing Eq. (8), determine the fragrance strength.

Choosing the top butterfly overall utilizing a^*

End for

Repeat

Allocate $it = it + 1$

For $(j = 1: j \leq m)$ do

Build the random numberqmc, where $qmc \in [0,1]$

If $qmc < \rho$ then//o is the perceived value

Butterflies travel in the direction of the best butterfly a^*utilizing Eq. (9)

Else

Butterfly movement should be random in Eq. (10)

End IF

Butterfly movement should be random $e\left(z_j^s\right)$

Choosing the top butterfly overall utilizing a^*

End for

Refresh the parametric values by utilizing Eq. (11)

Until $(it > max_{it})$

Create the finest possible results a^*

A notion or method known as multitemporal entails analyzing and contrasting different periods or temporal data. Multitemporal analysis is frequently used in several disciplines, including remote sensing, geography, and data analysis, to detect and comprehend alterations that take place over time. A multitemporal analysis is the process of gathering and comparing photographs taken several times in remote sensing and satellite photography. This makes it possible for scientists and analysts to investigate how land use, vegetation cover, urbanization, and other environmental factors have changed over time. Patterns, trends, and anomalies may be found by tracking modifications in these variables over time. This information is useful for evaluating the state of the environment and spotting changes.

In many different study domains, data classification and regression are crucial. By increasing the margin among modules in a high-dimensional space, supervised machine learning (ML) algorithms called SVMs are utilized to categorize data points. SVM is a powerful ML method utilized in various disciplines for data categorization, regression, and decision-making. Due to the technique's outstanding results in a variety of learning issues, which includes pattern identification, bioinformatics, and medical diagnosis, it has generated a lot of interest in the machine learning field. Employing a set of training data, an SVM may generate a regression framework or a classification function. This study uses the SVM as the classifier that seeks the ideal boundary among two classes. The highest division among decision classes is produced by the highest margin hyperplane. Support vectors are the training information that is most similar to the increased-margin hyperplane. When employing a training set containing input vectors and target labels, the SVM can resolve issues involving both linear and non-linear segmentations:

$$(v_1, z_j), j = 1 \cdots m, v\epsilon Q^m . z\epsilon\{+1, -1\}, v_j + a \geq +1 \; for \; z_j = +1 \tag{13}$$

Given that the conditions listed below are met:

$$v_j + a \leq -1 \; for \; z_j = +1$$

$$z_j(u_j \times u + a) - 1 \geq 0 \;\; \forall_j \tag{14}$$

That is equivalent to

$$u^{\times} V_j + a = 0 \tag{15}$$

This method is utilized to examine for a hyperplane $u \times v_j + a = 0$ to divide the information into classes of +1 and −1; it has the greatest margin in the feature space with a margin width among the hyperplanes equal to

$$\frac{2}{\|u\|^2} \tag{16}$$

The minimizing of the norm of u is the same as maximizing the margin. The classifier employs the decision function in primal weight space.

$$e(v) = sign(u \times v) + a \tag{17}$$

The subsequent optimization challenge was trained using the SVM:

$$Minimize: \frac{1}{2} U^S + D \sum_{j=1}^{M} \xi_j$$

$$z_j(u_j \times u + a) \geq 1 - \xi_j \; and \; \xi_j \geq 0 \tag{18}$$

Here D is a regularized value for slack variables that forces a trade-off among training error and generalization. These limitations are put in place to make sure that no training patterns fall outside of them. By removing noisy data employing the slack variables, the limitations are loosened. The classifier depicted in Eq. (8) was constrained since it only separates the data linearly. The limitation may be removed by projecting the input samples into a high-dimensional space, where a linear SVM may efficiently divide them. The mapping was carried out via kernel functions that created it possible to access higher-dimensional spaces without explicitly defining the function of the typically somewhat difficult mapping. By using recent data for training, the SVM may manage issues that cannot be solved linearly. After selecting various support vectors from the training set, the SVM then presents all data. A framework is created using the chosen support vectors after eliminating a few extreme values. In the beginning, binary classification issues were addressed by the SVM. SVM, on the other hand, has just lately been accepted for disease diagnosis, transition prediction, and treatment prognosis, and it makes use of both structural and functional neuroimaging data.

4. Result

The results of an intelligent COVID-19 monitoring system in the workplace can vary and can have a large effect on the overall safety of the workplace, the well-being of the employees, and organizational processes. Here, we have compared some of the traditional methods with our suggested techniques, they are artificial intelligence (AI [17]), Ensemble random forest (ERF [18]), and Internet of Things (IoT [19]).

In general, accuracy is the degree to which a measurement, forecast, or statement is accurate and close to the truth. The percent of right or accurate outcomes relative to all measurements or predictions is generally stated as a percentage or ratio when expressing accuracy. The reliability of the data, the predictive models employed, and the complexity and dynamics of the virus itself are some of the variables that may impact the COVID-19 monitoring forecasts' accuracy. It's crucial to highlight that because the COVID-19 pandemic is continually changing and a variety of external factors are at play, forecasting the exact future course of the disease is difficult. The ability of classifiers or machine learning systems to accurately predict category results is frequently measured by accuracy. It is calculated by dividing the total number of guesses by the number of accurate forecasts:

$$Accuracy \; = \; (Number \; of \; Correct \; Predictions \; / \; Total \; Number \; of \; Predictions) \; * \; 100 \tag{19}$$

In this equation, "Number of Correct Predictions" refers to the total number of occurrences that the framework or the framework properly predicted or classified, while "Total Number of Predictions" refers to all predictions produced.

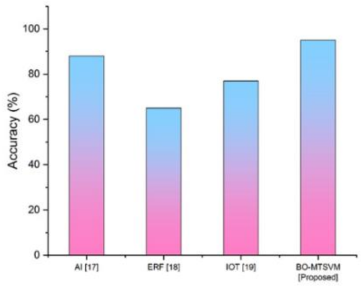

Fig. 2. Comparative analysis of accuracy with existing and suggested technique

Table 1. Numerical outcomes of accuracy

Methods	Accuracy (%)
AI [17]	88
ERF [18]	65
IOT [19]	77
BO-MTSVM [Proposed]	95

Figure 2 denotes the comparative analysis of accuracy with traditional and suggested techniques and Table 1 depicts the numerical outcomes of accuracy. When compared to other traditional methods, our suggested technique BO-MTSVM provides a high level for accuracy.

In particular for binary classification issues, precision is a performance indicator utilized to evaluate the accuracy of a classification model. It calculates the proportion of cases out of all displayed positive cases which were correctly predicted as positive (true positives). The following equation is used to determine precision:

$$Precision = (True\ Positives) / (True\ Positives + False\ Positives) \quad (20)$$

The term "True Positives" in the aforementioned formula indicates the number of occasions where the algorithm successfully anticipated a positive outcome, whereas the term "False Positives" denotes the number of instances where the system mistakenly projected a positive outcome.

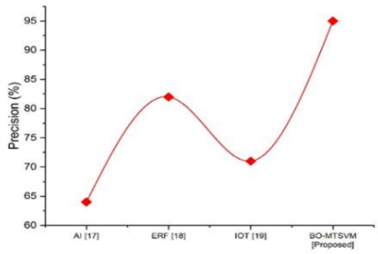

Fig. 3. Comparative analysis of precision with existing and suggested technique

Table 2. Numerical outcomes of precision

Methods	Precision (%)
AI [17]	64
ERF [18]	82
IOT [19]	71
BO-MTSVM [Proposed]	95

Figure 3 denotes the comparative analysis of precision with traditional and suggested techniques and Table 2 depicts the numerical outcomes of precision. When compared to other traditional methods, our suggested technique BO-MTSVM provides a high level for precision.

The time required to finish executing a program, job, or process is referred to as execution time. It is a measurement of the amount of time that has passed between the beginning and the finish of the execution, usually expressed in seconds or milliseconds. The equation that follows may be used, though, if you want to determine the execution time programmatically utilizing programming languages that include high-resolution timers or functions:

$$Execution\ Time = End\ Time - Start\ Time \quad (21)$$

The intricacy of the task, the effectiveness of the method or code, the hardware abilities of the system, the OS, and any external dependency are just a few of the variables that affect how long a program takes to execute.

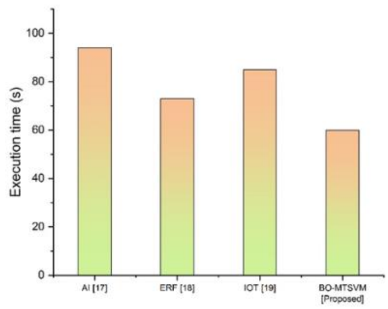

Fig. 4. Comparative analysis of execution time with existing and suggested technique

Table 3. Numerical outcomes of execution time

Methods	Execution time (s)
AI [17]	94
ERF [18]	73
IOT [19]	85
BO-MTSVM [Proposed]	60

Figure 4 denotes the comparative analysis of execution time with traditional and suggested techniques and table 3 depicts the numerical outcomes of execution time. When compared to other traditional methods, our suggested technique BO-MTSVM provides less time consumption while executing.

Sensitivity is a statistic employed to assess how well a binary classification model performs. It's also known as recall or true positive rate. It measures the proportion of real instances of positivity that the algorithm correctly identified. The equation that follows is used to determine sensitivity:

$$Sensitivity\ =True\ Positives\ /\ (True\ Positives\ + False\ Negatives) \quad (22)$$

In this equation, "True Positives" denotes the percentage of positive cases that were accurately predicted, while "False Negatives" is the percentage of negative examples that were mistakenly anticipated. Sensitivity is primarily relevant when the cost of false negatives is substantial and focuses on the model's capacity to accurately detect positive cases. It shows how accurately the model represents the real positive examples in the dataset.

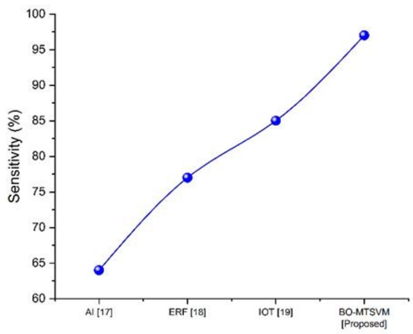

Fig. 5. Comparative analysis of sensitivity with existing and suggested technique

Table 4. Numerical outcomes of sensitivity

Methods	Sensitivity (%)
AI [17]	64
ERF [18]	77
IOT [19]	85
BO-MTSVM [Proposed]	97

Figure 5 denotes the comparative analysis of sensitivity with traditional and suggested techniques and Table 4 denotes the numerical outcomes of sensitivity. When compared to other traditional methods, our suggested technique BO-MTSVM provides a 97% high level for sensitivity.

5. Conclusion and Future Work

In summary, establishing a sophisticated COVID-19 monitoring system in the office setting can result in a number of important advantages. Companies can accomplish the following results by incorporating technology like temperature screening, symptoms tracking, contact tracing, occupant monitoring, and air quality monitoring. We introduced the BO-MTSVM approach, which is better than other traditional methods, 97% for sensitivity, 60% for execution time, 95% for precision, and 95% for accuracy in monitoring covid-19 system in a work environment. The future research is to create more sophisticated and effective monitoring systems and consider integrating emerging technologies including wearables, IoT sensors, AI, and machine learning (ML) algorithms. Examine the ways in which these technologies might improve the procedure of real-time data gathering, analysis, and decision-making.

References

[1] Singhal P, Srivastava PK, Tiwari AK, Shukla RK. A Survey: Wiryasaputra, R., Huang, C.Y., Kristiani, E., Liu, P.Y., Yeh, T.K. and Yang, C.T., (2022). Review of an intelligent indoor environment monitoring and management system for COVID-19 risk mitigation. Frontiers in public health, 10.

[2] Walters, M. and Kalinova, E., 2021. Virtualized care systems, medical artificial intelligence, and real-time clinical monitoring in COVID-19 diagnosis, screening, surveillance, and prevention. American Journal of Medical Research, 8(2), pp.37-50.

[3] Adhikari, M. and Munusamy, A., 2021. ICovidCare: Intelligent health monitoring framework for COVID-19 using ensemble random forest in edge networks. Internet of Things, 14, p.100385.

[4] Liang, H., Guo, Y., Chen, X., Ang, K.L., He, Y., Jiang, N., Du, Q., Zeng, Q., Lu, L., Gao, Z. and Li, L., 2022. Artificial intelligence for stepwise diagnosis and monitoring of COVID-19. European radiology, pp.1-11.

[5] Mijwil, M.M., Abttan, R.A. and Alkhazraji, A., 2022. Artificial intelligence for COVID-19: A short article. Artificial intelligence, 10(1).

[6] Rahman, M.Z., Akbar, M.A., Leiva, V., Tahir, A., Riaz, M.T. and Martin-Barreiro, C., (2023). An intelligent health monitoring and diagnosis system based on the Internet of things and fuzzy logic for cardiac arrhythmia COVID-19 patients. Computers in Biology and Medicine, 154, p.106583.

[7] Nasser, N., Emad-ul-Haq, Q., Imran, M., Ali, A., Razzak, I. and Al-Helali, A., (2021). A smart

healthcare framework for detection and monitoring of COVID-19 using IoT and cloud computing. Neural Computing and Applications, pp.1-15.

[8] Ahmed, S., Yong, J. and Shrestha, A., (2023). The Integral Role of Intelligent IoT Systems, Cloud Computing, Artificial Intelligence, and 5G in the User-Level Self-Monitoring of COVID-19. Electronics, 12(8), p.1912.

[9] Li, X., Wang, B., Liu, C., Freiheit, T. and Epureanu, B.I., (2020). Intelligent manufacturing systems in COVID-19 pandemic and beyond framework and impact assessment. Chinese Journal of Mechanical Engineering, 33, pp.1-5.

[10] Kallel, A., Rekik, M. and Khemakhem, M., (2021). IoT-fog-cloud based architecture for smart systems: Prototypes of autism and COVID-19 monitoring systems. Software: Practice and Experience, 51(1), pp.91-116.

[11] Taiwo, O. and Ezugwu, A.E., (2020). Smart healthcare support for remote patient monitoring during covid-19 quarantine. Informatics in medicine unlocked, 20, p.100428.

[12] Vedaei, S.S., Fotovvat, A., Mohebbian, M.R., Rahman, G.M., Wahid, K.A., Babyn, P., Marateb, H.R., Mansourian, M. and Sami, R., (2020). COVID-SAFE: An IoT-based system for automated health monitoring and surveillance in post-pandemic life. IEEE Access, 8, pp.188538-188551.

[13] Khan, M.M., Mehnaz, S., Shaha, A., Nayem, M. and Bourouis, S., (2021). IoT-based smart health monitoring system for COVID-19 patients. Computational and Mathematical Methods in Medicine.

[14] Otoom, M., Otoum, N., Alzubaidi, M.A., Etoom, Y. and Banihani, R., 2020. An IoT-based framework for early identification and monitoring of COVID-19 cases. Biomedical signal processing and control, 62, p.102149.

[15] Tavakoli, M., Carriere, J. and Torabi, A., 2020. Robotics, smart wearable technologies, and autonomous intelligent systems for healthcare during the COVID-19 pandemic: An analysis of the state of the art and future vision. Advanced Intelligent Systems, 2(7), p.2000071.

[16] Zhang, H., Cai, Y., Zhang, H. and Leung, C., 2020. A hybrid framework for smart and safe working environments in the era of COVID-19. Int. J. Inf. Technol, 26(1).

[17] Suri, J.S., Puvvula, A., Majhail, M., Biswas, M., Jamthikar, A.D., Saba, L., Faa, G., Singh, I.M., Oberleitner, R., Turk, M. and Srivastava, S., 2020. Integration of cardiovascular risk assessment with COVID-19 using artificial intelligence. Reviews in Cardiovascular Medicine, 21(4), pp.541-560.

[18] Adhikari, M. and Munusamy, A., 2021. ICovidCare: Intelligent health monitoring framework for COVID-19 using ensemble random forest in edge networks. Internet of Things, 14, p.100385.

[19] Bhardwaj, V., Joshi, R. and Gaur, A.M., 2022. IoT-based smart health monitoring system for COVID-19. SN Computer Science, 3(2), p.137.

[20] Mr. Anish Dhabliya. (2013). Ultra Wide Band Pulse Generation Using Advanced Design System Software . International Journal of New Practices in Management and Engineering, 2(02), 01 - 07. Retrieved from http://ijnpme.org/index.php/IJNPME/article/view/14

[21] Sharma, S. ., Kumar, N. ., & Kaswan, K. S. . (2023). Hybrid Software Reliability Model for Big Fault Data and Selection of Best Optimizer Using an Estimation Accuracy Function . International Journal on Recent and Innovation Trends in Computing and Communication, 11(1), 26–37. https://doi.org/10.17762/ijritcc.v11i1.5984

.