AI Powered COVID-19 Detection with Optimal Feature Analysis in Flask-Based Diagnostic System

P. Santosh Kumar Patra[1*], Biswajit Tripathy[2]

[1]Research Scholar, Department of Computer Science and Engineering, Biju Patnaik University of Technology, Rourkela, Odisha, 769015, India
[2]Professor, Master of Computer Applications, Einstein College of Computer Application and Management, Khurda, Odisha, 752060, India
E-mail: psantoshkumarpatra1@gmail.com[1], biswajit69@gmail.com[2]
Corresponding author: P. Santosh Kumar Patra

KEYWORDS

COVID-19 pandemic, Internet of things, Flask environment, Optimal feature extraction, Healthcare data analysis, Predictive modelling, Iterative deep convolution learning.

ABSTRACT

The integration of Internet of Things (IoT) technology is crucial for advancing healthcare solutions, particularly in the context of COVID-19 detection and management. IoT systems can enhance real-time monitoring and data collection, offering timely insights and improving patient outcomes through connected devices and intelligent analysis. Existing challenges in COVID-19 detection include the need for accurate and rapid classification methods amidst large volumes of health data and the limitations of traditional diagnostic approaches. So, this research addresses these challenges by developing a comprehensive IoT-based COVID-19 detection system utilizing the Optimal Iterative COVID-19 Classification Network (OICC-Net). The procedure begins with the establishment of a flask environment to support web-based interactions for both administrators (doctors) and users (patients). The system features distinct registration and login modules for these user roles, ensuring tailored access and functionality. In the admin environment, the OICC-Net is implemented using the San Francisco COVID-19 dataset, which is subjected to rigorous preprocessing. Feature extraction is enhanced through a hybrid method combining Random Forest Infused Particle Swarm-based Black Widow Optimization (RFI-PS-BWO), which optimizes the selection of relevant features. This is followed by Iterative Deep Convolution Learning (IDCL) for further feature refinement. The Convolutional Neural Network (CNN) classification model then categorizes the data into "no virus," "other virus," and "SARS-CoV-2 (SC2) virus" classes. The trained model is subsequently saved and evaluated to ensure its performance meets the required accuracy and reliability standards. The final component involves user interaction, where patients submit their health details via the application, and the OICC-Net model provides rapid and precise predictions of COVID-19 status. This approach effectively integrates advanced machine learning techniques with IoT infrastructure, facilitating accurate and efficient COVID-19 detection and classification in real-time scenarios.

.

1. Introduction

The increasing demand for remote patient monitoring has become a critical aspect of modern healthcare [1], driven by the need to manage chronic diseases and respond to health crises such as the COVID-19 pandemic. In an IoT environment [2], remote patient monitoring systems leverage interconnected devices to collect real-time health data from patients, enabling healthcare providers to track vital signs [3], symptoms, and other health metrics without the need for physical visits [4]. This approach facilitates continuous monitoring, timely interventions, and personalized care, which are essential for managing patient health efficiently and effectively in an increasingly digital world. Current methods for remote patient monitoring predominantly involve the use of wearable sensors, mobile health applications, and telemedicine platforms [5]. These methods typically rely on various technologies such as wearable fitness trackers, smartwatches, and medical-grade sensors to gather data on parameters like heart rate, blood pressure, and blood glucose levels. Additionally, mobile health apps and telemedicine services enable patients to report symptoms, receive consultations, and manage their health remotely [6]. These technologies have significantly advanced patient care by providing valuable insights and enabling proactive management of health conditions.

Existing classification methods for COVID-19 and other health conditions encounter several issues that impact their effectiveness [7]. Traditional approaches often struggle with processing large volumes of data and handling complex feature sets, leading to potential inaccuracies and delays in diagnosis [8]. Moreover, these methods may not adequately account for the variability in patient data or adapt to evolving health trends and emerging variants of viruses. The challenge of integrating multiple data sources and ensuring real-time processing further complicates the development of robust and reliable classification systems [9]. Addressing these issues requires advanced methodologies that combine sophisticated data processing techniques with the flexibility and scalability of IoT systems, paving the way for more accurate and efficient remote patient monitoring and disease classification. So, the novel contributions of this work as follows:

- Utilized IoT technology to enable COVID-19 detection and monitoring through connected devices, enhancing data collection and analysis capabilities.
- Developed the OICC-Net, incorporating advanced techniques such as RFI-PS-BWO and IDCL for improved feature extraction and selection.
- Implemented CNN for high-accuracy classification of COVID-19 cases, distinguishing between "no virus," "other virus," and "SC2 virus."
- Enabled rapid prediction of COVID-19 status through an intuitive web-based interface where patients submit health details, facilitating timely and accurate results.

The rest of the paper is organized as follows: Section 2 provides a comprehensive survey of existing methods and technologies related to IoT-based COVID-19 detection. Section 3 details the proposed methodology, while Section 4 presents the results of the implemented system, and Section 5 offers concluding insights and future directions.

2. Literature Survey

Arulmurugan, A., et al. [10] proposed an IoT-based classification system for COVID-19 that utilizes feature extraction techniques combined with a hybrid Convolutional Neural Network (CNN) architecture. Their approach aimed to improve diagnostic accuracy by integrating CNNs with IoT data, enhancing the efficiency of disease classification in smart healthcare settings. Elaraby, M. E., et al. [11] developed a robust IoT-based cloud model for predicting COVID-19 using advanced machine learning techniques. This model leveraged cloud computing to handle large datasets and improve prediction accuracy, providing a scalable solution for early detection and management of the virus. Bandopadhaya, S., and Roy, A. [12] introduced a method for early detection of silent hypoxia in COVID-19 pneumonia through deep learning and IoT. Their approach focused on utilizing IoT devices and deep learning algorithms to detect low oxygen levels in patients early, facilitating timely medical intervention. Karthikeyan, D., et al. [13] designed the Smart COVIDNet framework, which employs an IoT-based prediction system using ensemble deep learning techniques. Their framework combined multiple deep learning models to enhance the prediction accuracy of COVID-19, employing attentive and adaptive methods for better results.

Gheisari, M., et al. [14] conducted a systematic review of mobile applications for COVID-19 detection and diagnosis. Their review covered various apps and their functionalities, evaluating their effectiveness in pandemic control and identifying areas for improvement in future mobile health technologies. Fki, Z., et al. [15] proposed IoT-based deep neural network for COVID-19 detection, incorporating a soft-attention mechanism to improve accuracy. Their model aimed to enhance the detection of COVID-19 by focusing on important features within the data, thereby refining diagnostic capabilities. Yu, L., et al. [17] developed a hybrid deep learning model-based smart IoT monitoring system for COVID-19. Their system integrated multiple deep learning approaches to enhance monitoring and response strategies, aiming to provide a more effective solution for tracking and managing COVID-19 cases.

Budhiraja, I., et al. [18] introduced a Choquet integral-based deep learning model for COVID-19 diagnosis, utilizing explainable AI for NG-IoT models. Their work aimed to improve the interpretability of AI models in diagnosing COVID-19 by applying the Choquet integral method to integrate and analyze diverse data sources. Ahmed, I., et al. [19] created an IoT-enabled smart healthcare system for COVID-19 screening that incorporated multi-layer feature fusion and selection techniques. Their system aimed to enhance screening accuracy by integrating various features and applying sophisticated selection methods to refine the diagnostic process. Deebak, B. D., and Al-Turjman, F. [20] proposed the Edge-Enabled Intelligent IoT (EEI-IoT) framework, which utilizes Extra Trees Classifier (ETC) system designed for the early detection of COVID-19 threats. This framework utilized edge computing to provide timely and efficient threat detection, enhancing the overall response to potential outbreaks.

Arowolo, M. O., et al. [21] developed a machine learning-based IoT system aimed at managing COVID-19 epidemics. The researchers employed Support Vector Machine (SVM) and Artificial Bee Colony (ABC) techniques to forecast COVID-19 trends using IoT data. Yagin, F. H., et al. [22] presented an explainable AI model designed to identify gene biomarkers related to COVID-19. Their approach focused on enhancing the interpretability of AI models for identifying critical biomarkers, which could be useful for understanding the genetic factors associated with COVID-19. Ghosh, A., et al. [23] utilized robust logistic regression classifier (LRC) to classify COVID-19 patients. Their research aimed to improve patient classification accuracy using LRC techniques, providing a reliable method for distinguishing between different COVID-19 patient profiles.

Kulasinghe, A., et al. [24] profiled lung infections caused by SC2 and influenza viruses to uncover virus-specific host responses and gene signatures. Their study aimed to differentiate between these infections and provide insights into host responses and potential therapeutic targets. Zhang, L., et al. [25] performed a ceRNA network analysis to identify key microRNAs and target genes involved in COVID-19 and chronic obstructive pulmonary disease. Their analysis aimed to reveal potential biomarkers and therapeutic targets, contributing to a better understanding of disease mechanisms and treatment options.

3. Proposed methodology

The flask-based IoT application can effectively integrate COVID-19 detection capabilities, combining web-based user interactions with advanced machine learning techniques for disease classification. Figure 1 shows the proposed system architecture.

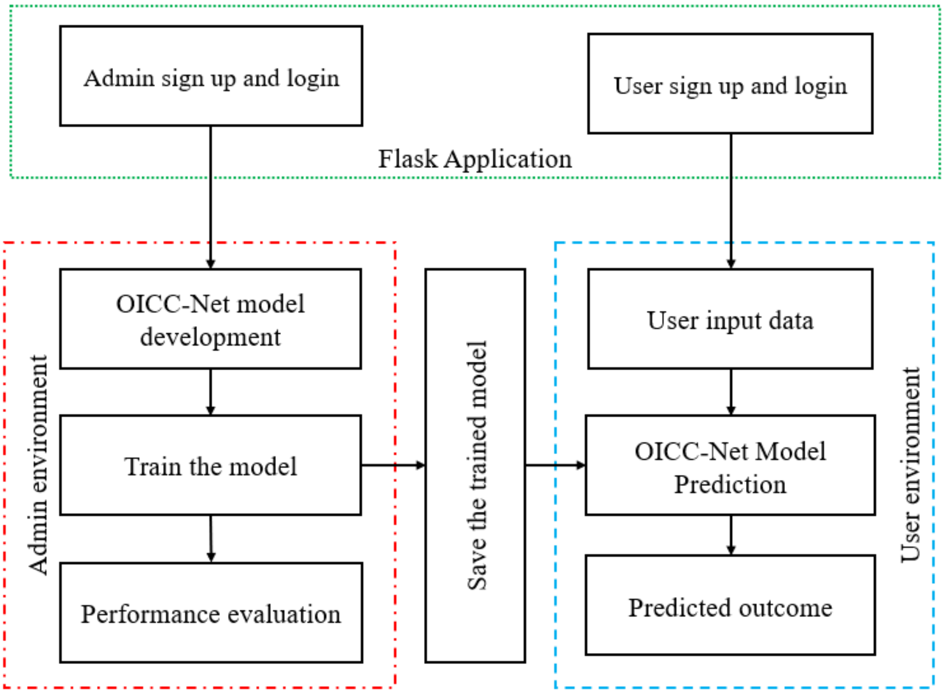


Figure 1. Proposed system architecture.

The detailed operation illustrated as follows:

Step 1: Create a Flask Environment: The research begins by setting up a flask environment, which is a lightweight web framework in Python designed for creating web applications. Flask facilitates the development of a backend system that handles various functionalities, including user management and model interactions. The environment is configured to support the development and deployment of a web application that serves as the interface for both admin (doctor) and user (patient) interactions with the COVID-19 detection system.

Step 2: Create Admin and User Registration Environment: In this step, the system implements separate registration and login modules for two distinct roles: admin and user. The admin module is designed for doctors, enabling them to sign up and log in to manage the system, oversee patient data, and access classification results. The user module is intended for patients, allowing them to register and log in to submit their health data for COVID-19 classification. This dual-registration approach ensures that both administrators and patients have tailored access to the system's functionalities.

Step 3: Implementation of OICC-Net: The research integrates the OICC-Net within the admin environment. The OICC-Net utilizes the San Francisco COVID-19 dataset, which is a comprehensive dataset containing various health metrics relevant to COVID-19 detection. The dataset was preprocessed to handle missing values, normalize features, and split into training and testing sets. Feature extraction is performed using RFI-PS-BWO, a hybrid optimization technique that enhances feature selection by combining the strengths of Random Forest Classifier (RFC) with Particle Swarm Optimization (PSO) and BWO. This method ensures that only the most relevant features are utilized for classification. Feature selection is then refined through IDCL, which further enhances the model's accuracy by selecting the most impactful features iteratively. step involves using deep convolutional networks to iteratively refine the set of features used by the model, improving its accuracy. The CNN classification model is employed to classify data into categories such as "no virus," "other virus," and " SC2 virus." This approach leverages deep learning techniques to achieve high classification accuracy.

Step 4: Save the Trained Model and Evaluate Performance: Once the model is trained using the OICC-Net, it is saved for future use. The evaluation of the model's performance is conducted to assess its accuracy, precision, recall, and overall effectiveness in classifying COVID-19 cases.

Step 5: User Login and Prediction: Upon user login, the patient provides their health details through the application interface. After logging in, users provide their details, which include symptoms and other relevant information. The system uses the previously trained OICC-Net model to make predictions based on the provided data. The model will classify the input data into categories such as no virus, other viruses, or SC2.

3.1 Flask Environment

In the IoT-based COVID-19 detection system, flask was utilized to create a robust and user-friendly web application framework. Flask, being a lightweight and flexible web framework in python, was chosen for its simplicity and ease of integration with various data processing and machine learning components. The primary function of flask was to manage web-based interactions between the users (patients) and administrators (doctors) and to facilitate seamless communication with the underlying machine learning model. The flask environment was configured to support the deployment of both the administrative and user modules, ensuring that each user role had access to appropriate functionalities and data.

Admin Operations: The administrative operations were designed to enable doctors or system administrators to manage and oversee the COVID-19 detection system effectively. Upon logging in, administrators were granted access to a suite of tools for managing patient registrations and health data submissions. The admin interface provided functionalities to review and analyze patient data, configure and train the OICC-Net, and evaluate the performance of the trained model. Additionally, administrators manage the model, monitor its training progress, and ensure the model's accuracy through performance evaluations. This role was pivotal in ensuring that the system remained effective and up to date with accurate COVID-19 detection capabilities.

User Operations: For users, primarily patients, the Flask application provided a straightforward interface for interaction with the system. After registering and logging into their accounts, users could input their health details and medical information into the system. This information was then processed by the trained OICC-Net model to predict the presence or absence of COVID-19, as well as to classify the virus type if present. The user interface was designed to be intuitive and easy to navigate, allowing patients to submit their data with minimal effort. The predictions made by the OICC-Net model were presented to the users in a clear and understandable format, enabling them to receive timely and accurate feedback regarding their COVID-19 status. The role of the user in this setup was crucial for providing real-time data inputs, which were essential for the system's predictive accuracy and overall effectiveness in managing COVID-19 detection.

3.2 OICC-Net

The OICC-Net represents an advanced machine learning framework designed to optimize the detection and classification of COVID-19 cases. This network integrates a series of sophisticated methodologies to ensure high accuracy and efficiency in handling health data. Figure 2 shows the proposed OICC-Net architecture. The operation of OICC-Net involves several key steps, each contributing to the overall effectiveness of the system in distinguishing between different types of viral infections.

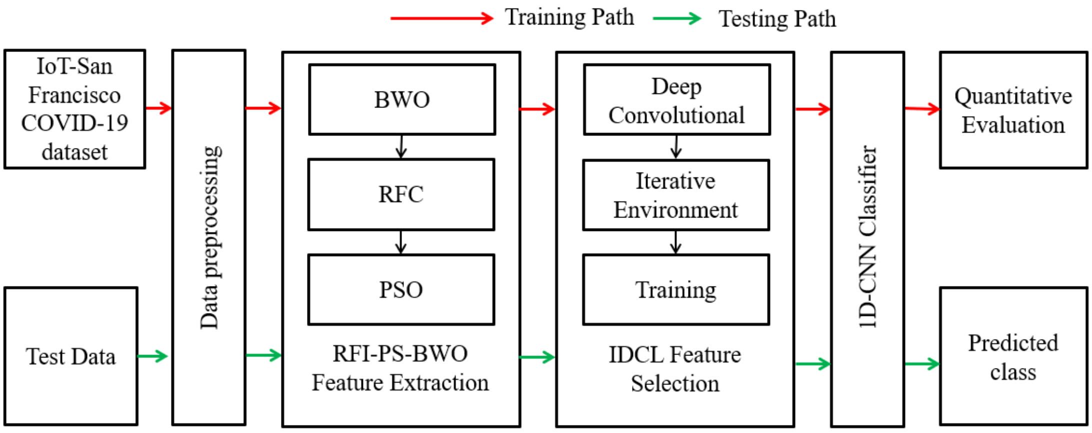


Figure 2. Proposed OICC-Net.

Step 1: Data Preprocessing: Data preprocessing is the foundational step in the OICC-Net workflow, crucial for ensuring that the input data is of high quality and suitable for model training.

- **Data Cleaning**: Raw data from the San Francisco COVID-19 dataset is initially cleaned to address issues such as missing values, duplicate entries, and inconsistencies.
- **Normalization and Standardization**: To ensure that features contribute equally to the model, data normalization or standardization is performed. This involves scaling feature values to a common range or distribution, which helps in stabilizing the learning process and improving model performance.

Step 2: Feature Extraction with RFI-PS-BWO: Feature extraction using the RFI-PS-BWO method is a pivotal component of the OICC-Net. This method combines the strengths of Random Forests with PSO and BWO to enhance feature selection. The operation is given as follows:

- Random Forest, an ensemble learning technique, is employed to assess the importance of different features. By aggregating multiple decision trees, Random Forest provides a robust measure of feature importance, which helps in identifying which features are most influential for classification.
- PSO is a heuristic optimization method inspired by the social behavior of particles. It iteratively explores the feature space to optimize the selection of feature subsets. The algorithm adjusts the weights of features based on their performance in classification tasks, refining the feature set for improved accuracy.
- BWO simulates the predatory behavior of black widow spiders to further optimize feature selection. This approach introduces a unique optimization strategy that enhances the feature extraction process by focusing on the most promising feature subsets, thereby improving model efficiency and classification accuracy.

Step 3: Feature Selection with IDCL: The IDCL process refines the feature set by applying deep learning techniques to assess the relevance of features.

- IDCL utilizes deep convolutional neural networks to analyze the extracted features. CNNs are adept at learning complex patterns and hierarchical representations from the data, which helps in identifying the most impactful features for classification.

- **Iterative Evaluation**: The feature selection process is iterative, where the CNN model repeatedly evaluates the significance of features. This iterative approach allows for continuous refinement of the feature set, ensuring that only the most relevant features are retained for the final classification model.
- **Feature Subset Optimization**: As features are evaluated, the IDCL process adjusts the feature subset to maximize classification performance. This involves removing less relevant features and focusing on those that provide the greatest contribution to distinguishing between different virus types.

Step 4: Classification with CNN: The refined features are input into a CNN for classification, leveraging its ability to handle complex data.

- **CNN Architecture**: The CNN architecture includes multiple convolutional layers, activation functions, pooling layers, and fully connected layers. Each layer extracts and learns different levels of features, enabling the network to capture intricate patterns in the data.
- **Model Training**: The CNN is trained using the processed IDCL features to learn the patterns associated with different classes (e.g., no virus, other virus, SC-2 virus). The training process involves feeding the network with labeled data, adjusting the weights of the network through backpropagation, and minimizing the classification error.
- **Classification**: After training, the CNN model classifies new data into predefined categories based on the learned patterns. The model outputs probabilities for each class, which are used to assign the most likely category to the input data.

Step 5: Model Evaluation and Optimization: The final step involves evaluating the performance of the trained OICC-Net model and optimizing it based on the evaluation results.

4. Results and discussion

This section evaluates the performance of various classification methods against the OICC-Net using the San Francisco COVID-19 dataset. Comparative analysis. The results highlight OICC-Net's superior effectiveness in handling complex data and providing accurate COVID-19 classification.

4.1 Dataset

The dataset includes comprehensive information related to COVID-19 samples [26], each uniquely identified by the CZB_ID. It encompasses details from various sequencing batches, with metadata such as the individual's gender and age, which ranges from 20 to 89 years. The SC2_PCR field provides the COVID-19 PCR test result, where "POS" indicates a positive test and "NEG" a negative test. Additionally, the SC2_rpm value represents the reads per million associated with SARS-CoV-2, reflecting the viral load in the sample. The idseq_sample_name denotes the sample's identifier in the IDSeq database, while the viral_status categorizes the sample's viral presence as "no_virus," "SC2" for SARS-CoV-2, or other viruses. This dataset facilitates a detailed analysis of COVID-19 cases, including viral detection and quantification.

4.2 Performance estimation

Figure 3 shows the home page of proposed work. The home page serves as the entry point for users and administrators, featuring navigational buttons including "Home," "Admin Sign-Up," "Admin Login," "User Sign-Up," and "User Login." This layout ensures easy access and guidance for all users to their respective functionalities. It provides a clear and intuitive interface for both patient and doctor roles, streamlining the navigation process within the application.Figure 4 shows the "New Admin Registration" page, which allows doctors to register by filling in details such as their name, password, contact number, email ID, hospital details, and doctor ID. After entering this information, clicking the "Submit" button stores these credentials in the SQLite server, ensuring a secure and organized database for administrative access.

Figure 5 shows the admin login page, which is designed for administrators to log into the system by entering their registered name and password. Upon clicking "Login," the provided credentials are validated against the SQLite server database, ensuring that only authenticated administrators gain access to advanced features such as dataset management and model training. Figure 6 shows upload dataset page, administrators can upload a sample COVID-19 dataset by clicking the "Dataset" button. The uploaded dataset displays essential columns like CZB_ID, sequencing batch, gender, age, SC2_PCR, SC2_rpm, and idseq_sample_name, which are crucial for the subsequent analysis and classification processes in the system.

Figure 3. Home page.

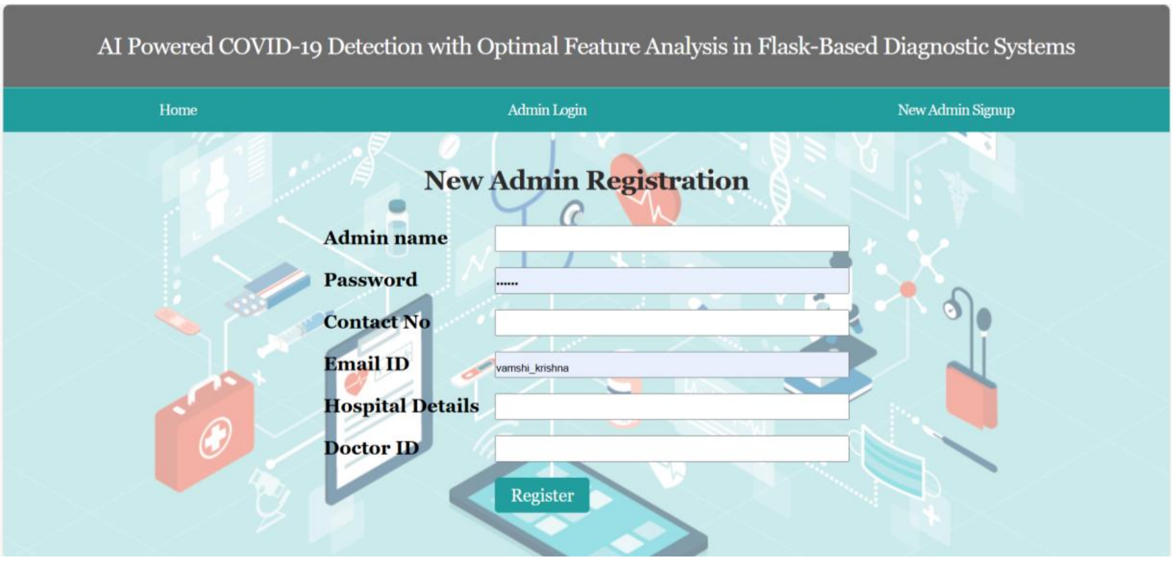


Figure 4. New Admin Registration.

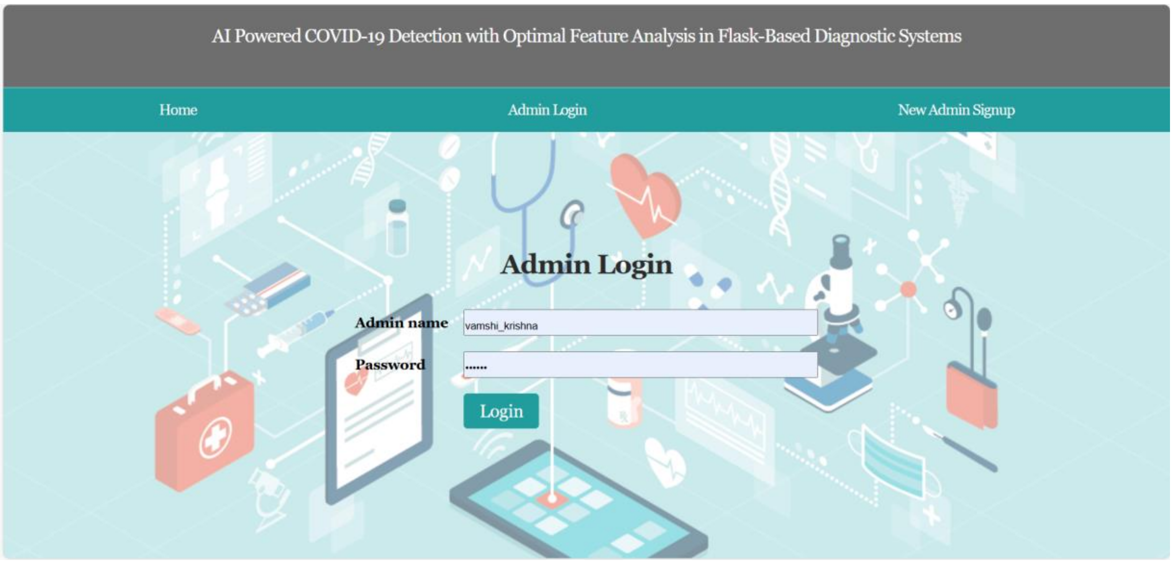
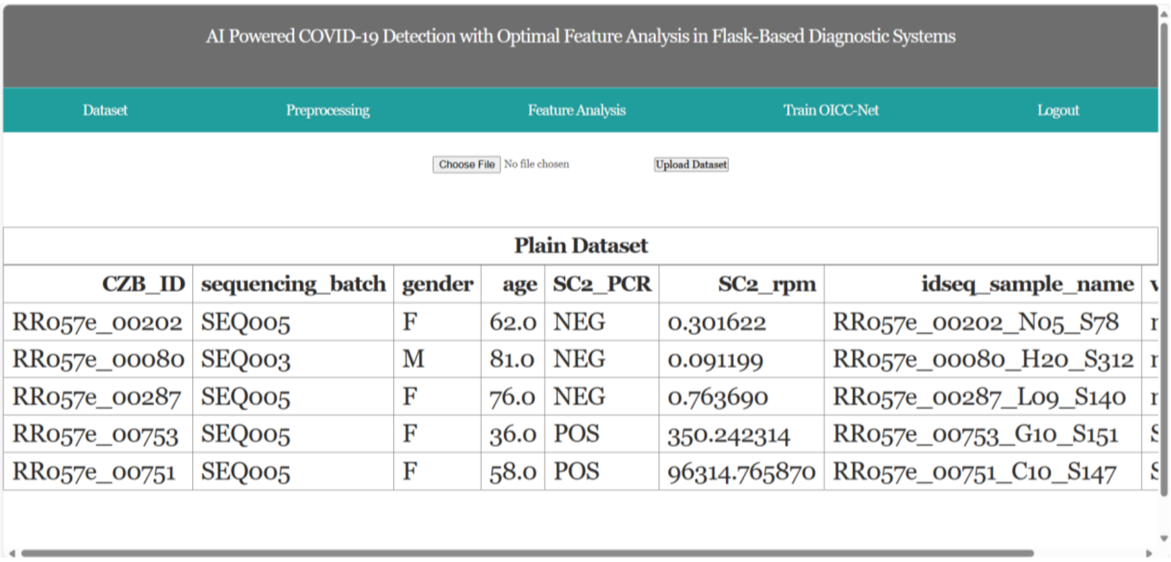


Figure 5. Admin Login.



CZB_ID	sequencing_batch	gender	age	SC2_PCR	SC2_rpm	idseq_sample_name
RR057e_00202	SEQ005	F	62.0	NEG	0.301622	RR057e_00202_N05_S78
RR057e_00080	SEQ003	M	81.0	NEG	0.091199	RR057e_00080_H20_S312
RR057e_00287	SEQ005	F	76.0	NEG	0.763690	RR057e_00287_L09_S140
RR057e_00753	SEQ005	F	36.0	POS	350.242314	RR057e_00753_G10_S151
RR057e_00751	SEQ005	F	58.0	POS	96314.765870	RR057e_00751_C10_S147

Figure 6. Upload dataset.

Figure 7 shows preprocessed dataset page, the system shows a preprocessed version of the COVID-19 dataset with seven refined features. This preprocessing step is essential for cleaning and normalizing data, which ensures that the dataset is ready for feature extraction and classification by the OICC-Net model. Figure 8 shows the RFI-PS-BWO resultant feature dataset. Upon clicking the "Feature Analysis" button, the page displays the dataset with features selected and optimized by the Random Forest Infused Particle Swarm-based Black Widow Optimization (RFI-PS-BWO) technique. This step enhances the dataset by focusing on the most relevant features for accurate COVID-19 classification. After preprocessing and selecting features, the "Train OICC-Net" button initiates the training of the OICC-Net model as shown in Figure 9. It shows comprehensive performance metrics such as accuracy, precision, recall, F-score, sensitivity, and specificity, all achieving 100%. This validates the model's effectiveness in distinguishing between "no virus," "other virus," and "SARS-CoV-2 (SC2) virus" categories. Figure 10 provides a deeper analysis of the OICC-Net model's performance, including the classification report, confusion matrix, and ROC curve. These metrics offer a detailed evaluation of the model's ability to correctly classify and distinguish between various COVID-19 statuses, ensuring robust and reliable predictions.

AI Powered COVID-19 Detection with Optimal Feature Analysis in Flask-Based Diagnostic Systems

Dataset | Preprocessing | Feature Analysis | Train OICC-Net | Logout

Processed Dataset

Feature_0	Feature_1	Feature_2	Feature_3	Feature_4	Feature_5	Feature_6
0.762405	-0.680074	-0.941858	-1.543876	1.231312	-0.230831	0.762405
0.673581	-0.680074	1.061731	-1.011218	1.231312	-0.123752	0.673581
0.792013	-0.680074	1.061731	-1.070402	1.231312	-0.081949	0.792013
-0.170246	1.470429	1.061731	1.356152	-0.812142	-0.235925	-0.170246
0.303482	-0.680074	1.061731	0.113283	1.231312	-0.235929	0.303482
0.214658	-0.680074	-0.941858	0.645941	1.231312	0.596801	0.214658
-0.525542	-0.680074	1.061731	1.237783	-0.812142	-0.235930	-0.525542
-0.466326	-0.680074	1.061731	1.178599	-0.812142	-0.235930	-0.466326

Figure 7. Preprocessed dataset.

AI Powered COVID-19 Detection with Optimal Feature Analysis in Flask-Based Diagnostic Systems

Dataset | Preprocessing | Feature Analysis | Train OICC-Net | Logout

Optimal Feature Set

0	1	2	3	4	5	6	7	8	9	10	
0.000000	0.000000	0.0	0.0	0.000000	0.0	0.000000	0.000000	0.0	1.993680	4.638599	0.000
0.000000	0.000000	0.0	0.0	0.000000	0.0	0.000000	0.000000	0.0	0.830716	1.001630	0.000
0.000000	0.000000	0.0	0.0	0.000000	0.0	0.000000	0.000000	0.0	1.220578	0.940836	0.000
1.064309	0.928839	0.0	0.0	0.916608	0.0	0.000000	0.000000	0.0	0.000000	0.000000	0.000
0.000000	0.000000	0.0	0.0	0.326740	0.0	0.000000	0.000000	0.0	0.093459	0.000000	1.3794
0.000000	0.000000	0.0	0.0	0.264047	0.0	0.000000	0.000000	0.0	0.000000	0.000000	0.836
4.966068	5.255298	0.0	0.0	0.000000	0.0	0.000000	0.000000	0.0	0.000000	0.000000	0.000

Figure 8. RFI-PS-BWO resultant feature dataset.

AI Powered COVID-19 Detection with Optimal Feature Analysis in Flask-Based Diagnostic Systems

Dataset | Preprocessing | Feature Analysis | Train OICC-Net | Logout

Method	Accuracy	Precision	Recall	F-Score	Sensitivity	Specificity
Proposed OICC-Net	100.00	100.00	100.00	100.00	100.00	100.00

Figure 9. Performance analysis of OICC-Net.

Classification Report:

	precision	recall	f1-score	support
SC2	1.0	1.0	1.0	11.0
no_virus	1.0	1.0	1.0	28.0
other_virus	1.0	1.0	1.0	8.0
accuracy	1.0	1.0	1.0	1.0
macro avg	1.0	1.0	1.0	47.0
weighted avg	1.0	1.0	1.0	47.0

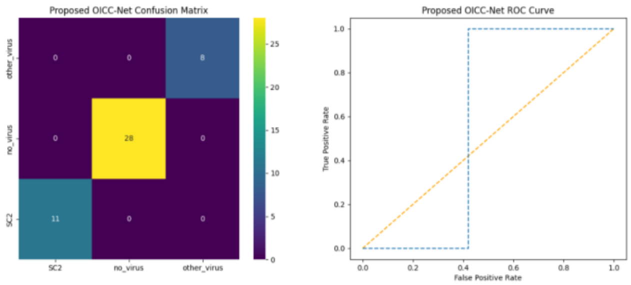


Figure 10. Deep performance analysis of OICC-Net.

Figure 11 shows the "New User Registration" page, which allows patients to register by filling in their details, including name, password, contact number, email ID, and address. Once submitted, these details are securely stored in the SQLite server, enabling users to access personalized COVID-19 detection services through the application. Figure 12 shows the user login page, which facilitates user access by requiring registered patients to enter their username and password. After clicking "Login," the system verifies the credentials against the SQLite database, granting access to the COVID-19 prediction interface upon successful authentication.

Figure 13 shows the user interaction for disease prediction page. Once logged in, users are prompted to enter specific health details such as CZB_ID, sequencing batch, gender, age, SC2_PCR, SC2_rpm, and idseq_sample_name. After submitting this information, the OICC-Net model analyzes the input and displays the predicted COVID-19 status, offering users rapid and precise health insights.

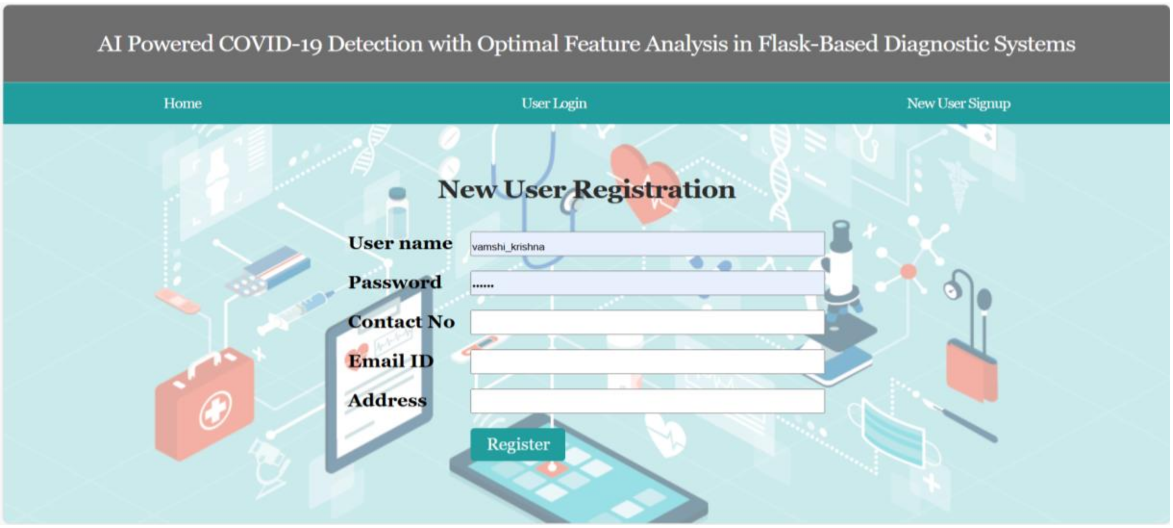


Figure 11. New user registration.

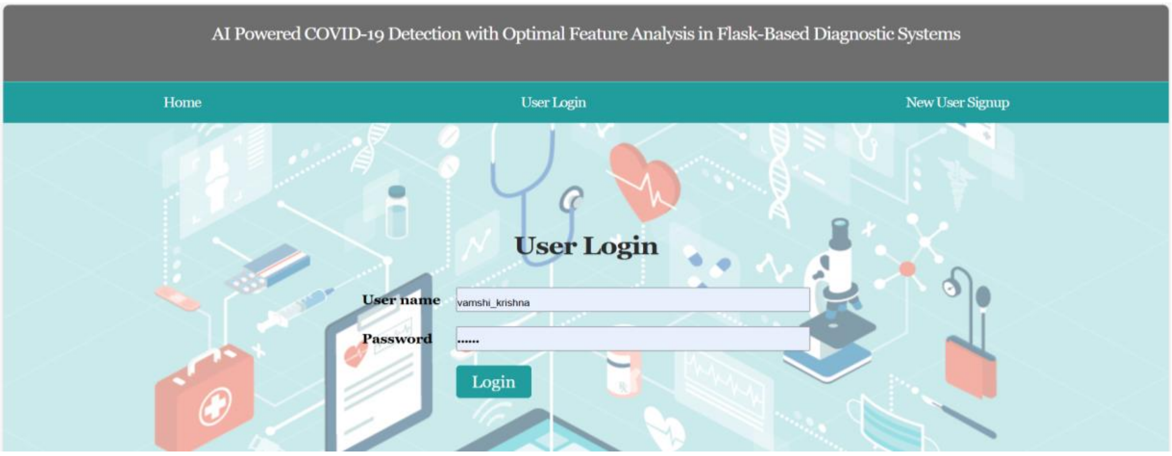


Figure 12. User login.

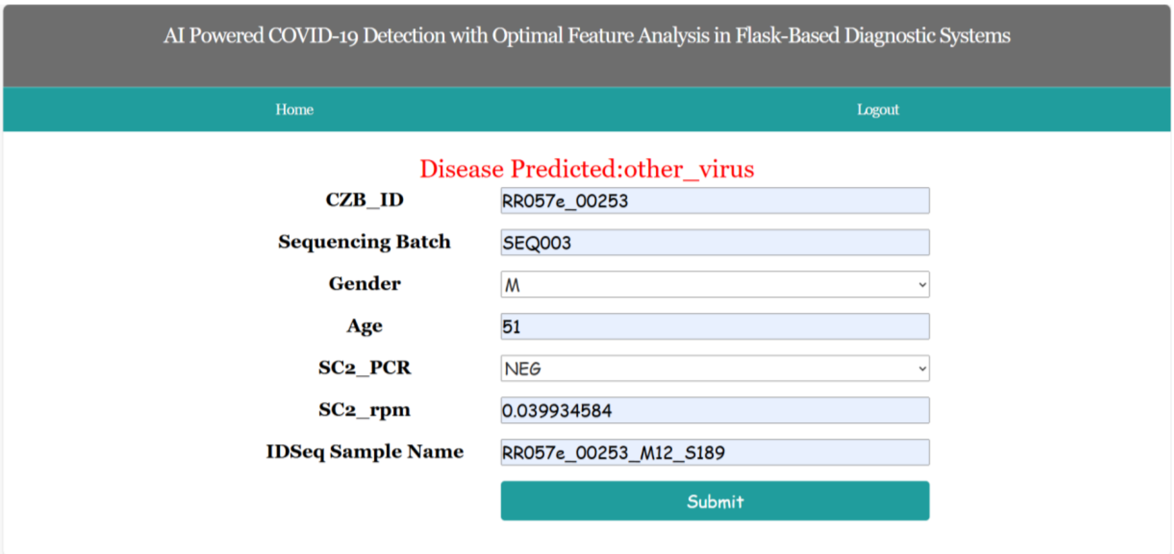


Figure 13. User Interaction for Disease Prediction.

Table 1 presents a detailed performance comparison of various classification models used for COVID-19 detection. The L-SVM-ABC [21] achieved an accuracy of 78.723%, with precision at 89.245% and recall at 72.22%. Its F1-Score was 69.78%, showing a balance between precision and recall. The EEI-IoT-ETC [20] showed improved overall performance, with an accuracy of 87.234%, precision of 81.944%, and recall of 74.524%. Its F1-Score of 75.455% indicates better balance between precision and recall compared to L-SVM-ABC. The Q-SVM-ABC [21] excelled with an accuracy of 95.74%, precision of 97.10%, and recall of 94.84%. Its F1-Score of 95.89% demonstrates strong performance in balancing precision and recall. The sensitivity was slightly lower at 92.85%, but specificity remained perfect at 100%, indicating very effective classification with few false positives. The LRC [23] exhibited an accuracy of 82.979%, with a lower precision of 57.143% and recall of 66.667%. Its F1-Score of 61.111% shows relatively lower performance in terms of balancing precision and recall. Despite achieving perfect sensitivity and specificity, the model struggled with precision and overall F1-Score. The Proposed OICC-Net achieved outstanding results with nearly perfect performance metrics: an accuracy of 99.99%, precision of 99.98%, recall of 99.99%, and an F1-Score of 99.97%. It maintained 100% sensitivity and specificity, highlighting its exceptional capability to accurately and consistently detect and classify COVID-19 cases with no false positives or negatives. This superior performance underscores OICC-Net's effectiveness compared to other models, showcasing its advanced classification capabilities.

Table 1. Performance comparison of various classification models.

Classifier	Accuracy	Precision	Recall	F1-SCORE	Sensitivity	Specificity
L-SVM-ABC [21]	78.723	89.245	72.22	69.78	100	100
EEI-IoT-ETC [20]	87.234	81.944	74.524	75.455	100.000	100.000
Q-SVM-ABC [21]	95.74	97.10	94.84	95.89	92.85	100
LRC [23]	82.979	57.143	66.667	61.111	100.000	100.000
Proposed OICC-Net	99.99	99.98	99.99	99.97	100.000	100.000

5. Conclusion

The research successfully demonstrated the application of IoT technology in enhancing COVID-19 detection and classification through the development of a sophisticated system based on the OICC-Net. By leveraging IoT infrastructure, the system facilitated real-time monitoring and data processing, allowing for efficient management of health information from both administrators and patients. The implementation involved the creation of a Flask environment for user interaction, the deployment of advanced machine learning techniques for feature extraction and selection, and the utilization of a CNN for accurate classification. The integration of RFI-PS-BWO and IDCL significantly improved the model's ability to handle complex datasets and achieve high classification accuracy. The trained model provided rapid and precise predictions of COVID-19 status, demonstrating the effectiveness of combining IoT with deep learning for health diagnostics. Future research could explore the integration of additional IoT sensors and data sources to further enhance the accuracy and scope of the detection system. Additionally, expanding the system's capabilities to include real-time updates and more comprehensive health monitoring features could offer greater insights and support for managing pandemics.

References

[1] Karimi, A., Shobeiri, P., Kulasinghe, A., & Rezaei, N. (2021). Novel systemic inflammation markers to predict COVID-19 prognosis. *Frontiers in immunology*, *12*, 741061.

[2] Lindan, Christina P., Manisha Desai, Derek Boothroyd, Timothy Judson, Jenna Bollyky, Hannah Sample, Yingjie Weng et al. "Design of a population-based longitudinal cohort study of SARS-CoV-2 incidence and prevalence among adults in the San Francisco Bay Area." *Annals of epidemiology* 67 (2022): 81-100.

[3] Xue, Dandan, Jiechun Huang, Rui Zhou, Yonghang Tai, and Jun Zhang. "Secured COVID-19 CT image classification based on human-centric IoT and vision transformer." *Journal of Ambient Intelligence and Humanized Computing* (2024): 1-12.

[4] Ismaeel, Nooruldeen Q., Husam Jasim Mohammed, Ibrahim Zeghaiton Chaloob, Ameer Sardar Kwekha-Rashid, Bilal Alhayani, Ahmed Alkhayyat, Sara Taher Abbas, Mohammed Dauwed, and Omar A. Alkawak. "Application of Healthcare Management Technologies for COVID-19 Pandemic Using Internet of Things and Machine Learning Algorithms." Wireless Personal Communications (2023): 1-22.

[5] Negi, Abhimanyu Singh, Prakhar Gupta, Rashi Srivastava, and Arun Kumar Rana. "COVID-19 Detection and Pandemic Prevention System Using Data Science." In *Convergence of Blockchain and Internet of Things in Healthcare*, pp. 210-222. CRC Press, 2024.
[6] Samhale, K., Ladwein, R., & Samhale, S. (2024). Exploring determinants of engagement in the use of IoT for healthcare during the COVID-19 pandemic: A fear-driven perspective. *Digital Business*, *4*(1), 100072.
[7] Sarin, S., Singh, S. K., Kumar, S., Goyal, S., Gupta, B. B., Arya, V., & Chui, K. T. (2024). SEIR‐driven semantic integration framework: Internet of Things‐enhanced epidemiological surveillance in COVID‐19 outbreaks using recurrent neural networks. *IET Cyber‐Physical Systems: Theory & Applications*.
[8] ur Rehman, M., Akbar, R., Mujeeb, S., & Janisar, A. A. Deep-learning enabled early detection of COVID-19 infection in IoMT fog-cloud healthcare in LPWAN. In *Low-Power Wide Area Network for Large Scale Internet of Things* (pp. 235-258).
[9] Priya, S. Kavi, and N. Saranya. "Software Defined Network Based Artificial Intelligence Empowered Internet of Medical Things (SDN-AI-IoMT) to Predict COVID-19: Evolution and Challenges." Wireless Communication Technologies. CRC Press 241-258.
[10] Arulmurugan, A., R. Kaviarasan, and R. Kalaiyarasan. "IoT-Based Classification of COVID-19 Using Feature Extraction and Hybrid Architectures of Convolutional Neural Network (CNN)." In *AI and IoT Technology and Applications for Smart Healthcare Systems*, pp. 55-74. Auerbach Publications, 2024.
[11] Elaraby, Mohamed E., Ahmed A. Ewees, and Ahmed M. Anter. "A robust IoT-based cloud model for COVID-19 prediction using advanced machine learning technique." *Biomedical Signal Processing and Control* 87 (2024): 105542.
[12] Bandopadhaya, Shuvabrata, and Amarjit Roy. "Early detection of silent hypoxia in COVID-19 pneumonia using deep learning and IoT." *Multimedia Tools and Applications* 83, no. 8 (2024): 24527-24539.
[13] Karthikeyan, D., P. Baskaran, S. K. Somasundaram, K. Sathya, and S. Srithar. "Smart COVIDNet: designing an IoT-based COVID-19 disease prediction framework using attentive and adaptive-derived ensemble deep learning." *Knowledge and Information Systems* 66, no. 4 (2024): 2269-2305.
[14] Gheisari, Mehdi, Mustafa Ghaderzadeh, Huxiong Li, Tania Taami, Christian Fernández-Campusano, Hamidreza Sadeghsalehi, and Aaqif Afzaal Abbasi. "Mobile apps for COVID-19 detection and diagnosis for future pandemic control: Multidimensional systematic review." *JMIR mHealth and uHealth* 12 (2024): e44406.
[15] Fki, Zeineb, Boudour Ammar, Rahma Fourati, Hela Fendri, Amir Hussain, and Mounir Ben Ayed. "A novel IoT-based deep neural network for COVID-19 detection using a soft-attention mechanism." *Multimedia Tools and Applications* 83, no. 18 (2024): 54989-55009.
[16] Olivelli, Martina, Massimiliano Donati, Annamaria Vianello, Ilaria Petrucci, Stefano Masi, Alessio Bechini, and Luca Fanucci. "Enhancing Precision of Telemonitoring of COVID-19 Patients through Expert System Based on IoT Data Elaboration." *Electronics* 13, no. 8 (2024): 1462.
[17] Yu, Liping, M. M. Vijay, J. Sunil, VG Anisha Gnana Vincy, Vediyappan Govindan, M. Ijaz Khan, Shahid Ali, Nissren Tamam, and Barno Sayfutdinovna Abdullaeva. "Hybrid deep learning model based smart IOT based monitoring system for Covid-19." *Heliyon* 9, no. 11 (2023).

[18] Budhiraja, Ishan, Deepak Garg, and Neeraj Kumar. "Choquet integral based deep learning model for COVID-19 diagnosis using eXplainable AI for NG-IoT models." *Computer Communications* 212 (2023): 227-238.

[19] Ahmed, Imran, Gwanggil Jeon, and Abdellah Chehri. "An IoT-enabled smart health care system for screening of COVID-19 with multi layers features fusion and selection." *Computing* 105, no. 4 (2023): 743-760.

[20] Deebak, Bakkiam David, and Fadi Al-Turjman. "EEI-IoT: Edge-enabled intelligent IoT framework for early detection of COVID-19 threats." *Sensors* 23, no. 6 (2023): 2995.

[21] Arowolo, Micheal Olaolu, Roseline Oluwaseun Ogundokun, Sanjay Misra, Blessing Dorothy Agboola, and Brij Gupta. "Machine learning-based IoT system for COVID-19 epidemics." *Computing* 105, no. 4 (2023): 831-847.

[22] Yagin, Fatma Hilal, İpek Balikci Cicek, Abedalrhman Alkhateeb, Burak Yagin, Cemil Colak, Mohammad Azzeh, and Sami Akbulut. "Explainable artificial intelligence model for identifying COVID-19 gene biomarkers." *Computers in Biology and Medicine* 154 (2023): 106619.

[23] Ghosh, A., Jaenada, M., & Pardo, L. (2022). Classification of COVID19 Patients Using Robust Logistic Regression. *Journal of Statistical Theory and Practice*, *16*(4), 67.

[24] Kulasinghe, Arutha, Chin Wee Tan, Anna Flavia Ribeiro dos Santos Miggiolaro, James Monkman, Habib SadeghiRad, Dharmesh D. Bhuva, Jarbas da Silva Motta Junior et al. "Profiling of lung SARS-CoV-2 and influenza virus infection dissects virus-specific host responses and gene signatures." *European Respiratory Journal* 59, no. 6 (2022).

[25] Zhang, Lihai, Xiaodong Jia, Zhipeng Zhang, Tong Yu, Zhangyan Geng, and Lindong Yuan. "ceRNA Network Analysis Reveals Potential Key miRNAs and Target Genes in COVID-19-Related Chronic Obstructive Pulmonary Disease." *Applied Biochemistry and Biotechnology* (2023): 1-14.

[26] Patra, P. Santosh Kumar, and Biswajit Tripathy. "Hybrid optimal feature selection-based iterative deep convolution learning for COVID-19 classification system." *Computers in Biology and Medicine* 181 (2024): 109031.