International Journal of Computational Methods and Experimental Measurements
https://www.acadlore.com/journals/IJCMEM

MIMIC-EYE: A Secure and Explainable Multi-Modal Deep Learning Framework for Clinical Decision Support

Santosh Kumar[1,2*], Shaik Sagar Imambi[1]

[1] Computer Science & Engineering, KL Education Foundation (Deemed to be University), 522302 Vaddeswaram, India
[2] Artificial Intelligence & Data Science, Vishwakarma Institute of Technology, 411037 Pune, India

* Correspondence: Santosh Kumar (santosh.kumar@viit.ac.in)

Received: 4-16-2025 **Revised:** 5-19-2025 **Accepted:** 5-25-2025

Citation: S. Kumar and S. S. Imambi, "MIMIC-EYE: A secure and explainable multi-modal deep learning framework for clinical decision support," *Int. J. Comput. Methods Exp. Meas.*, vol. 13, no. 3, pp. 484–506, 2025. https://doi.org/10.56578/ijcmem130303.

Abstract: The integration of heterogeneous medical data remains a major challenge for clinical decision support systems (CDSS). Most existing deep learning (DL) approaches rely primarily on imaging modalities, overlooking the complementary diagnostic value of electronic health records (EHR) and physiological signals such as electrocardiograms (ECG). This study introduces MIMIC-EYE, a secure and explainable multi-modal framework that fuses ECG, chest X-ray (CXR), and MIMIC-III EHR data to enhance diagnostic performance and interpretability. The framework employs a rigorous preprocessing pipeline combining min–max scaling, multiple imputation by chained equations (MICE), Hidden Markov Models (HMMs), Deep Kalman Filters (DKF), and denoising autoencoders to extract robust latent representations. Multi-modal features are fused through concatenation and optimized using a Hybrid Slime Mould–Moth Flame (HSMMF) strategy for feature selection. The predictive module integrates ensemble DL architectures with attention mechanisms and skip connections to capture complex inter-modal dependencies. Model explainability is achieved through Local Interpretable Model-agnostic Explanations (LIME) and Shapley Additive Explanations (SHAP), enabling transparent clinical reasoning. Experimental results demonstrate superior performance, achieving 98.41% accuracy, 98.99% precision, and 98.0% sensitivity—outperforming state-of-the-art baselines. The proposed MIMIC-EYE framework establishes a secure, interpretable, and generalizable foundation for trustworthy AI-driven decision support in critical care environments.

Keywords: EHR and ECG signals; Explainable AI; HSMMF; LIME and SHAP; Multi-model DL

1 Introduction

Diagnosing abnormalities and illnesses from medical images, like chest X-rays, CT scans, or MRIs, is known as medical image diagnostics [1]. The radiologists and doctors are among the specialists with the necessary skills and knowledge to carry out this task. Nonetheless, data from 2019 indicated a global deficit of healthcare professionals of, 30.6 million, 6.4 million, and 2.9 million, respectively [2]. The problem has gotten worse since healthcare personnel have been disproportionately affected by the pandemic. AI-enabled diagnostic systems have emerged as promising ways to assist clinicians, lessen workload, and sustain accurate diagnostic throughput when resources are limited [3]. Deep Learning (DL) technologies have shown to be highly accurate for diagnostic tasks, including chest X-ray classification for pneumonia and pulmonary nodule and arrhythmia detection from ECG signals [4], and have been shown to match human expert levels of performance [5, 6]. DL models are inspired by biological neural networks and learn directly from large datasets, and their performance is dependent on the quality and diversity of the training data [7]. In this context, our study describes a multi-modal DL framework (MIMIC-Eye) that integrates image, signal, and EHR data to automate and improve clinical decision-making and provide a service that can compensate for the shortage of healthcare workers.

In addition to giving DL models more reliable data, it can be advantageous to expose other properties of the instances to the approach. Three well-liked DL methods are contrastive learning, multi-tasking, and multi-modal learning [8]. To improve the scenarios' description or extension to models, a range of modalities are used, which improves the scenarios' adaptability and performance [9]. Multiple input modalities are used in multimodal learning,

which helps the model better understand the scenario and see different facets of the input phenomena [10]. The information gathered from each sensor is then referred to as "modality". Multi-modal data refers to the information gathered when multiple sensors are positioned to view a phenomenon. The system is trained by a variety of tasks that call for different kinds of labels in multi-task learning [11]. A framework will adapt to each task and perform better in generalization when it is acquired from multiple tasks. Using a self-supervised method called contrastive learning, a model is trained to compare and contrast inputs by mapping distinct modes to the similar meaningful vector space [12, 13]. Each of these methods makes use of the extra data provided by the range of modalities.

The dataset employed in this study is the Medical Information Mart for Intensive Care (MIMIC) dataset [14]. Numerous subsets of the MIMIC dataset have been created to offer more details and modalities in response to the dataset's increased popularity. It is essential to incorporate two types of extra data in the study. The patient's clinical information is the first kind. For radiologists to make an accurate diagnosis, clinical data is extremely instructive and necessary [15, 16]. When radiologists are diagnosing patients, they collect data that is centered around people. Experts possess the competence of medical diagnosis; thus, it is advantageous to research and examine their patterns of diagnosis. It can investigate the procedures that radiologists use to analyze medical images for human-centric data [17].

This work offers MIMIC-Eye, a safe and intelligible multi-modal DL framework that leverages federated learning to safeguard patient security and privacy and enhance clinical decision services. The model employs multi-modal learning to integrate new knowledge from ECG signals, chest X-ray images, and EHR data to adequately capture the complete view of each patient's condition. The model also employs multi-task learning, whereby the model jointly performs related predictive tasks (e.g., disease detection and risk scoring), which can improve the generalisation capabilities of the model. Finally, it leverages contrastive learning during the feature-representation learning step to increase class separability while minimising redundant features. Concatenation and model-based data fusion techniques are used for data fusion following the application of hybrid optimization algorithms, such as the slime mould and moth flame algorithm (HSMMF) for feature selection. To forecast the health condition, ensemble DL approaches with skip connections and attention layers are all combined. Explainable AI (XAI) techniques that combine LIME and SHAP models are utilized to ensure Decision Support system (DSS). The key contributions of the MIMIC-Eye framework are as follows:

• MIMIC-Eye integrates diverse data sources, specixfically, MIMIC-III EHR data, ECG, and chest X-ray, enhancing the depth and breadth of clinical insights.

• Relevant features are extracted using specialized techniques to ensure that the most informative features from each dataset are utilized.

• Combines the HSMMF to enhance model performance.

• Employ a concatenation and modal-based data fusion technique to ensure patient data privacy and security during the data fusion process.

• Ensemble DL Approaches is developed with Skip Connections and Attention Layers to improve the accuracy and reliability of health condition forecasts.

• LIME and SHAP Models combined to provide transparent and interpretable DSS enhancing the trust and acceptance of medical professionals in the system.

The organization of the paper is followed as; section 2 and section 3 detailed related works and problem definition, section 4 elaborates the proposed methodology, section 5 discusses gained results and outcomes, and section 6 ends with a conclusion.

2 Related Works

To lessen the need for manual annotations and hence lower training costs, Ma et al. [18] present the Eye-gaze Guided Multi-modal Alignment (EGMA) model, which uses eye-gaze data to improve image and text feature alignment. The developed method has strong performance, surpassing other cutting-edge techniques in zero-shot categorization and retrieval assignments. Including readily available eye-gaze data in regular radiological diagnosis represents a step toward reducing reliance on manual annotation.

Hsieh et al. [19] develop DL applications in medical imaging that solely employ image data. In this work, it provides a dataset called MIMIC-EYE, which is an extensive integration of several MIMIC-related information. Incorporating data from eye tracking with several MIMIC technologies may provide a better understanding of physicians' visual searching patterns and the development of more reliable, precise, and consistent DL methods for medical image detection.

Kim et al. [20] suggest a novel way to improve the interaction between people and computers in chest X-ray processing utilizing Vision-Language (VL) systems enriched with radiologists' attention. The suggested method makes use of heatmaps created from eye gaze data, which superimpose on medical images to identify regions of great interest for radiologists while evaluating chest X-rays. Furthermore, eye gazing was shown to have a positive effect on fine-tuning, as it exceeded other medical VLMs in all tests except answering visual questions.

Hsieh et al. [21] introduced EyeXNet, a multimodal DL system that predicts abnormality spots in chest X-rays by integrating images and radiologists' fixation masks. Because radiologists tend to concentrate on abnormally shaped regions and offer predictive algorithms with more specific locations to work with, the developed technique concentrates on fixation maps between reporting moments. Also, compare fixations made by radiologists during quiet and reporting times and find that attachments are more specific and targeted after reporting.

Hsieh et al. [22] presented a unique architecture made up of two fusion algorithms that allow the model to analyze the image data of clinical data, and chest X-rays from patients simultaneously. Also suggest the spatialization approach as a means of spatial organization to support the multimodal learning procedure in a Mask R-CNN approach, given that the modalities of the data are situated in distinct dimensional regions. The findings demonstrate that the suggested fusion approaches enhance the localization of illness in chest X-rays by about 12% based on Average Precision.

Yin et al. [23] concentrate on creating a patient stratification DL algorithm that can recognize and elucidate specific patient groups using multimodal EHRs. Here, create a topic-modeling-based Distinct Multimodal Learning System for EHR (ConMEHR). To create a single image space and diversify patient groups, ConMEHR uses contrastive learning (CL) processes at the modality and topic levels, respectively. The two real-world EHR datasets will be used to assess ConMEHR's performance, and the findings demonstrate better performance.

Sharif et al. [24] propose a novel automated DL technique for multiclass brain tumor categorization. The Densenet201 Pre-Trained DL Network is trained and refined to utilize a deep transfer of unbalanced data knowledge to implement the suggested strategy. Two feature selection strategies are proposed for a clear classification. A changed genetic algorithm based on the optimization technique is the second method. Ultimately, a multiclass SVM cubic classifier is used to classify the fused EKbHFV and MGA-based features, which, utilizing a distinct serial-based method, were merged.

Gaw et al. [25] suggest a multi-modality-based diagnostics DSS (MM-DDS). Three interconnected components make up MMI-DDS: a principal component analysis (PCA) to co-register multimodality data, a novel classifier for controlled particle swarm optimization (PSO) from all visualization modalities, and a clinical usefulness engine that uses opposite operations to find imaging features that contribute to the diagnosis of the disease. MMI-DDS demonstrates a much higher level of diagnostic accuracy when compared to the use of single imaging modalities alone.

Current DL models for clinical decision support are not interpretable, generalise poorly, and process single-modality inputs, which is clinically of little use. To address these issues, MIMIC-Eye combines ECG, chest X-ray, and EHR input to enhance adaptability, employs LIME and SHAP explainability, and ensemble DL with attention for precision. Federated learning maintains data privacy during training and is thus deployable in practice.

3 Problem Definition

Synchronizing cross-modal features is a major difficulty in multi-modal frameworks. Most emphasis on multi-modal pre-training techniques on either global or specific alignment within modalities and employ large datasets. This bottom-up approach frequently has interpretability issues, which pose a serious radiological risk [26]. Radiologists have to evaluate medical images with confidence in their work by utilizing a variety of patient data. Unfortunately, due to a lack of research on medical datasets mixing many data modalities, the majority of DL applications in medical imaging solely employ image data. Arranging and integrating diverse information across several modalities—which includes both organized and unstructured data—is the main problem [27]. In addition, while many computer vision approaches have been put forth that aim to increase classification accuracy, most of these approaches have weaknesses with respect to adaptability, transparency, and clinical relevance. For this reason, none of the existing attempts has transitioned into an applicable, interpretable, and secure clinical decision support system.

To tackle these gaps, this research proposes MIMIC-Eye, explainable federated multi-modal DL framework, which together utilize ECG, chest X-ray and EHR data. It seeks to address data heterogeneity, improve model interpretability, protect patient privacy and provide trustworthy diagnosis predictions that are also transparent.

4 Proposed Methodology

In the current healthcare environment, clinical DSS (also known as CDSS) are crucial for assisting doctors in making well-informed judgments. The architecture of the developed model is shown in Figure 1.

Nevertheless, there are numerous challenges in combining different data sources, such as Electronic Health Records (EHR) and eye tracking. This work offers MIMIC-Eye, a safe and intelligible multi-modal DL framework that leverages federated learning to safeguard patient security and privacy and enhance clinical DSS. MIMIC-Eye integrates MIMIC-III EHR data, ECG data, and chest X-ray data using advanced preprocessing techniques like probabilistic imputation for EHR data, deep Kalman filters for blurring chest X-ray image data, and bandpass filters for ECG data. Using a feature extraction method the domain-specific features and relevant features from each dataset are extracted. Concatenation and model-based data fusion techniques are used for data fusion following the application of hybrid optimization algorithms, such as the HSMMF for feature selection. To forecast the health condition, ensemble

DL approaches with skip connections and attention layers are all combined. Explainable AI (XAI) techniques that combine LIME and SHAP models are utilized to ensure DSS. By ensuring explainability and security and improving clinical decision accuracy, the proposed MIMIC-Eye architecture fosters medical professionals' acceptability and trust.

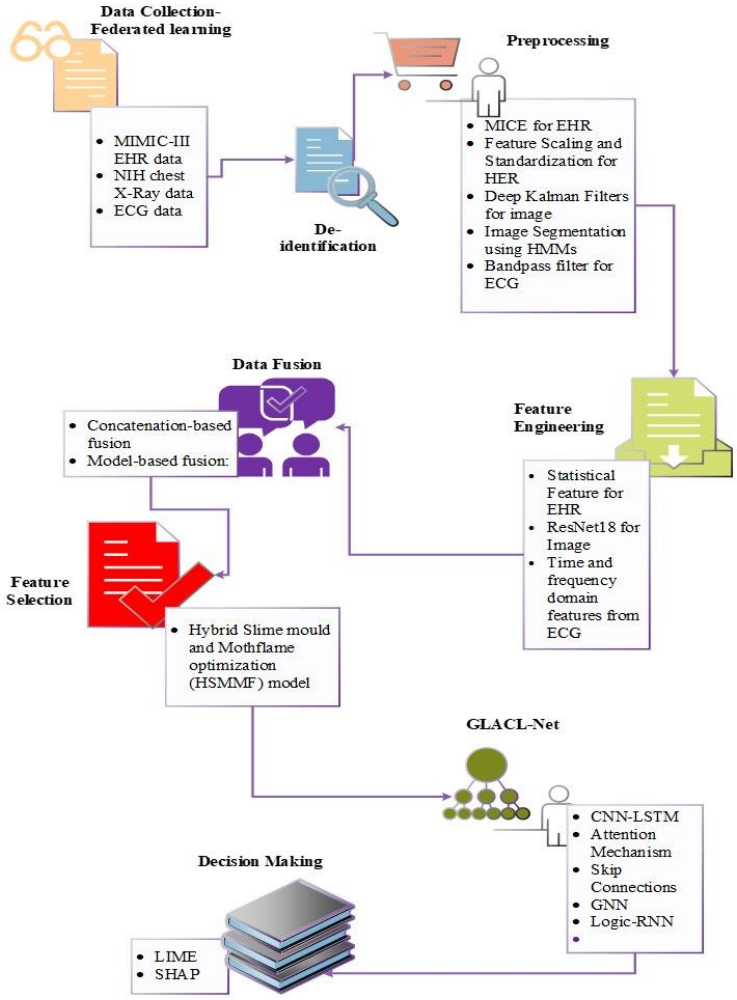

Figure 1. Developed framework architecture

4.1 Data Collection-Federated Learning

This refined workflow integrates MIMIC-III EHR data, ECG data, and NIH Chest X-ray images a DL model for improved DS in healthcare.

MIMIC-III EHR data [28]: It is still unclear what factors influence the in-hospital mortality of heart failure (HF) patients hospitalized in intensive care units (ICUs). Create and verify a prediction framework for ICU-admitted heart failure patients' all-cause in-hospital mortality. Data on demographics, vital signs, and laboratory values were taken from the following tables in the MIMIC III dataset: PATIENTS, D_LABIEVENTS, ADMISSIONS, NOTEEVENTS, ICUSTAYS, DIAGNOSIS_ICD, LABEVENTS, D_ICD DIAGNOSIS, CHARTEVENTS, D_ITEMS, and OUTPUTEVENTS. These characteristics provide the contextual and historical clinical data necessary to understand patient condition and outcomes.

Table 1. ECG data collection

Parameters	Arrhythmia Dataset	PTB Diagnostic ECG Database
Samples	109446	14552
Categories	5	2
Statistical Frequency	125 Hz	125 Hz

NIH chest X-ray data [29]: There are 30,805 distinct patients' X-ray images (112, 120) with disease labels included in this NIH Chest X-ray Dataset. The labels are intended to be appropriate for weakly-supervised learning and to be more than 90% correct. These images provide spatial information about thoracic pathology such as pneumonia, nodules, and infiltrates.

ECG data [30]: This dataset consists of two distinct sets of cardiac signals extracted from two popular databases for heartbeat categorization: the MIT-BIH Arrhythmia Dataset and the PTB Diagnostic ECG Database, which is detailed in Table 1. Both collections include enough number of samples to train a deep neural network. There are several CSV files in this collection. These CSV files are composed of matrices, where each row is an example from that subset of the dataset. Every part of these preprocessed and segmented data represents a heartbeat. This modality captures electrical signals, over a period of time, from the heart to identify arrhythmias and extract patterns of cardiac function that can be used to predict different types of diseases.

Each modality adds to the prediction in its distinct way: EHR provides context, ECG provides physiological signals, and X-rays provide structural abnormalities. All three modalities combine uniquely in the MIMIC-Eye to provide predictions that are more accurate, data-rich, clinically relevant and reliable.

4.2 De-Identification

In the context of sensitive patient data processing, de-identification is an important step, especially for effective modeling and healthcare research. By anonymization, direct identifiers such as names, social security values, and other private details are eliminated or hidden from patient data. By doing this, it is ensured that specific patients cannot be easily identified, safeguarding their privacy. Without jeopardizing the anonymity of patients, de-identification preserves the data's usefulness and enables investigators and programmers to use it efficiently for building and training DL models. This procedure is necessary to support the development of healthcare innovations while adhering to moral and regulatory requirements.

4.3 Preprocessing

Probabilistic Imputation Techniques MICE for HER

Multiple imputations using chained equations (MICE) [31] are a technique for imputing missing values that use correlations with other variables and are estimated based on the distribution of comparable data points. The imputation method known as MICE bases the missing data estimate on other variables present in the dataset. Every variable with values that are lacking under MICE is first subjected to a distinct univariate conditional model created by the analyst. Next, when undertaking imputation, the analyst designates an order in which to cycle through the conditional algorithms' sequence. The variables are listed as $\left(Z_{(1)}, \ldots\ldots, Z_{(p)}\right)$. Subsequently, the analyst sets each $Z_{mis(j)}$. The most common choices are to take a sample from (i) the correspondence's marginal probability $Z_{obs(j)}$ or (ii) the distribution conditional of $Z_{(j)}$ considering every other variable, created utilizing only available situations. The MICE method cycles through the series of univariate algorithms via an iterative procedure that begins with initialization. Regarding every variable j at each iteration t, one matches the conditional $\left(Z_{(j)} \left| Z_{obs(j)}, \left\{ Z_{(n)}^{(t)} : n < j \right\}, \left\{ Z_{(n)}^{(t-1)} : n > j \right\} \right.\right)$. Next, one takes the place of utilizing the model that is suggested $\left(Z_{mis(n)}^{(t)} \left| Z_{obs(j)}, \left\{ Z_{(n)}^{(t)} : n < j \right\}, \left\{ Z_{(n)}^{(t-1)} : n > j \right\} \right.\right)$.

The process of iteration persists for T completed dataset consists of all iterations through convergence including the values at the last iteration. $Z^{(l)} = \left(Z_{obs}, Z_{mis}^{(T)}\right)$. The entire process is then repeated l times to create the l completed datasets.

EHR Feature Scaling and Standardization

To guarantee that features are on the same scale and enhance model convergence, normalize features utilizing min-max scaling. By subtracting the minimum value from the data and dividing the result by the range—that is, the difference among the maximum and minimum—min-max normalization is applied using Eq. (1).

$$p^* = \left[\frac{p - \min(p)}{[\max(p) - \min(p)]} \right] \tag{1}$$

Let, $\min(p)$ is denoted as a minimum; $\max(p)$ is denoted as maximum; and p is considered as the distinction between the minimum and maximum. The interval's length is one, and the range is contained inside the interval [0,1].

Deep Kalman Filters for Image Data Denoising

Since Kalman Filters (KF) [32] can reduce the associated variance under certain ideal model assumptions, it is a popular approach for estimating the states of dynamic systems. A state Eqs. (2) and (3) can be used to characterize the behavior the system demonstrates in a discrete-time environment:

$$a_t = X a_{t-1} + Y e_{t-1} + f_{t-1} \tag{2}$$

$$b_t = Za_t + g_t \tag{3}$$

Let, a_t is to be projected for the state vector, a_{t-1} and e_{t-1} are the input vectors and the state from the prior time step, and b_t symbolizes the vector of measurement. X and Y are the matrix systems and Z is the matrix of measurements. The vectors f_{t-1} and g_t are connected to the noise of measurement & the procedure of additive noise, which are taken to be the zero-mean Gaussian operations. The first forecasting step is determined in Eqs. (4) and (5) and the update phase in Eqs. (6)-(8) yield the final estimate.

$$a'_t = Xa_{t-1} + Ye_{t-1} \tag{4}$$

$$S'_t = XS_{t-1}X^T + R \tag{5}$$

$$k_t = S'_t Z^T \left(ZS'_t Z^T + Q\right)^{-1} \tag{6}$$

$$a_t = a'_t + k_t \left(b_t - Za'_t\right) \tag{7}$$

$$S_t = \left(I - k_t Z\right) S'_t \tag{8}$$

The X-posterior state estimate, a_t is acquired by combining the linear X-prior estimate a'_t and the residual, which is a weighted distinction between the actual & the projected measurements in Eq. (7); the Kalman gain k in Eq. (6) minimizes the a-posteriori error covariance S in Eq. (8), that is established by the user. Lastly, R and Q are measurements of noise and the processing covariance matrices. R simulates the uncertainty of dynamics, while Q depicts the internal sounds of the sensor. These matrices significantly impact the final filter performance, so accurately estimating noise statistics requires a challenging tuning procedure. For sensors to estimate biases, effective fine-tuning is also crucial.

Denoising autoencoder: A deep convolutional architecture called DAE can be used to improve clear, unaltered results from input data that has been partially contaminated. The input data is purposefully tampered with in the original method via stochastic mapping in Eq. (9).

$$\tilde{a} = Q_d \left(\frac{\tilde{a}}{a}\right) \tag{9}$$

The tainted input is then mapped, much like in the case of a conventional autoencoder, to a hidden depiction in Eq. (10).

$$H = f_\varphi(\tilde{a}) = p(G\tilde{a} + y) \tag{10}$$

Ultimately, a reconstructed signal corresponds back to the hidden depiction in Eq. (11).

$$\tilde{a} = G\varphi.(H) \tag{11}$$

To reduce the Γ^2 reconstruction error, the resultant signal is contrasted with a reference signal throughout the training process using Eq. (12).

$$\Gamma(a - \tilde{a}) = \|a - \tilde{a}\|^2 = \|a - p(G\tilde{a} + y)\|^2 \tag{12}$$

The ground truth values supplied by the dataset serve as a reference signal, and the noisy angle prediction made by the KF is fed into the DKF.

Using Hidden Markov Models (HMMs) for Image Segmentation

For this work, a Hidden Markov represent (HMM) [33] based technique is adopted to represent the sequential pattern of human gait. A double-embedded stochastic procedure, of which one is hidden since it can only be viewed through a second stochastic procedure, is what defines an HMM. This random procedure is described using Markov chains. The best match between the supplied data and labels is achieved by selecting the HMM model. The Gaussian mixture models (GMMs) were used to explain the hidden states in the emission patterns for every HMM. The predicted heterogeneity of various transitions and stride parameters, which can be anticipated in data from free-living subjects, were chosen to be modeled by GMMs. Unsupervised training is done on HMMs. Design of the

Stride Pattern Every concealed state corresponded to a stride's subphase. The total number of states in the framework determined the time granularity for these subphases. This indicates that a monotonic rising state sequence is used to convey each step $S(0)$ to $S(n)$ considering the odds of self-transition $P(n,n)$ considering the likelihood of a move to the nearby state $P(n,n+1)$. Every single stride is taken out of the training dataset's image dataset to initialize and train the stride model. This is produced in k training protocols for k labelled progress. To begin, hidden states are created by naively segmenting each step into nn sections with equal spacing to determine the starting parameters for n GMMs, with the quantity of concealed states. All edges that already existed in the transition matrix were equally initialized. The segmented output results are shown in Figure 2.

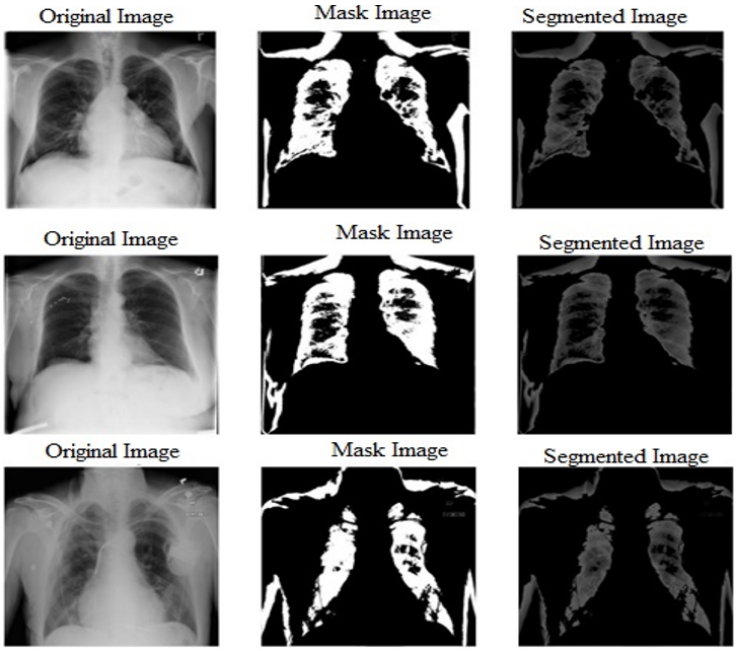

Figure 2. Segmented output results

Bandpass Filter for ECG Data

For the ECG, the first filtering technique's result is noise reduction utilizing the bandpass filter, which consists of a low-pass filter transmitted by a high-pass filter. A low-pass filter is used to reduce high-frequency interference. The use of digital filters with numerical coefficients in the design of filters enables real-time processing capacities. Speed is high since floating point processing is not needed. Eq. (13) illustrates the transfer function of the second-class low-pass filter.

$$k(x) = \left(1 - x^{-6}\right)^2 / \left(1 - x^{-1}\right)^2 \tag{13}$$

The filter has an 11Hz cut-off frequency, a 5-sample delay, and a 36dB gain. Eq. (14) displays the filter's variance equation.

$$h(nt) = 2h(nt - t) - h(nt - 2t) + g(nt) - 2g(nt - 6t) + g(nt - 12t) \tag{14}$$

An initial order low pass filter is subtracted from a complete pass filter with delay to create the high pass filter. Eq. (15) displays the high pass filter's transmission function.

$$kKp(x) = P(x)/G(x) = k^{-16} - kLp(k)/32 \tag{15}$$

Finally, it is attained as Eq. (16) illustrates.

$$kKp(x) = \left(-k^{32} + 32k^{16} - 32k^{15} + 1\right) / \left(32k^{32} - 32k^{31}\right) \tag{16}$$

Eq. (17) displays the bandpass filter's differential equation.

$$p(nt) = g(nt - 16t) - 0.03125[h(nt - t) + g(nt) - g(nt - 32t)] \tag{17}$$

The filter has a delay of 80 msec and a low cut-off rate of roughly 5Hz. Unity is the gain.

4.4 Feature Engineering

Statistical Feature Extraction for EHR

Extracting statistical features from EHR involves summarizing the raw data into meaningful statistics that can be used for further analysis or modeling. This process typically involves calculating various statistical measures that capture the data distribution's central tendency, dispersion, and shape. The statistical features and their mathematical explanations:

Mean: The total of all values split by the total number of values yields the mean. It offers a central tendency measurement using Eq. (18).

$$\mu = \frac{1}{n}\sum_{i=1}^{n} R_i \tag{18}$$

Let, R_i depicts every data point and n is the overall quantity of information.

Median: While values are arranged in an ascending order in a collection of data, the term "median" refers to the middle value. The sum of the two center values is the median when there are an even quantity of values.

Variance: The variation of the data points surrounding the mean is measured by variance. It is calculated as the mean of the squared deviations from the mean using Eq. (19).

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n} (R_i - \mu)^2 \tag{19}$$

Standard Deviation (SD): A metric used to determine dispersion in the precise same units as the data, the SD is the square root of the variance using Eq. (20).

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (R_i - \mu)^2} \tag{20}$$

Skewness: The imbalance of the data distribution is measured by skewness, which is measured using Eq. (21).

$$S_k = \frac{1}{n}\sum_{i=1}^{n} \left(\frac{R_i - \mu}{\sigma}\right)^3 \tag{21}$$

Minimum value: For every patient, determine the lowest value of a particular clinical variable for a given length of time.

Maximum value: For every patient, determine the maximum value of a particular clinical variable during a given length of time.

Risk score: Calculate a risk score by combining various clinical variables. To calculate risk ratings for new patients, use the logistic regression model that has been trained. The method used to determine the risk score using Eq. (22).

$$R(s) = \frac{1}{1 + e^{-(\delta_0 + \delta_1 R_1 + \delta_2 R_2 + \ldots.. + \delta_n R_n)}} \tag{22}$$

Let, $R(s)$ indicates the likelihood that the patient has the relevant condition, δ_0 is denoted as intercept, and $\delta_2 R_2 + \ldots\ldots + \delta_n R_n$ are the characteristics' coefficients $R_1, R_2, \ldots R_n$.

ResNet for Extracting Features from Chest X-Ray Image Data

CNNs that are used for deep feature extraction include the ResNet-18 framework [34]. ResNet-18 is a member of the ResNet-xx network group. A ReLU layer, an average pooling layer, five convolutional layers for gathering deep feature maps, and a fully connected layer for transforming feature maps from two dimensions to one as well as categorizing all input images characterized by feature vectors through the appropriate class of ResNet-18 network's eighteen deep layers. Finally, gather the features from the chest X-Ray images using the softMax activation function. The ResNet-18 model's architecture, with numerous layers and about 11.5 million parameters, is depicted in Figure 3.

Time and Frequency Domain Features from ECG Data

Time Domain Features: The time-based features such as R, P, S, and Q waves, heart rate, and RR interval are extracted from ECG data. The intervals between the ECG signal's subsequent R waves. It's a basic indicator of heart rate fluctuation or HRV. The heart rate, expressed in beats per minute (BPM), is the quantity of heartbeats per unit of time. Length of the ECG signal's P, T, QRS complex, and S waves. The ECG signal's RR interval is the interval of time between consecutive R waves. Flexibility in RR intervals can reveal information about the overall health of the heart and the activity of the autonomic nervous system.

Frequency Domain Features: When predicting Power Spectral Density (PSD), which has sharp peaks at the anticipated frequencies, the Pisarenko approach is especially helpful. The polynomial $B(f)$. This can then be used to calculate the PSD since it has zeros on the unit circle in Eq. (23).

$$B(f) = \sum_{n=0}^{m} b_k e^{-j2\rho fn} \tag{23}$$

Let, $B(f)$ symbolizes the intended polynomial, b_k symbolizes the required polynomial's coefficients, and m symbolizes the eigen filter's order $B(f)$. The Pisarenko approach uses the eigenvector that corresponds to the smallest eigenvalue (p_{pis}) and generates the desired polynomial's signal PSD using Eq. (24).

$$p_{pis}(f) = \frac{1}{|B(f)|^2} \tag{24}$$

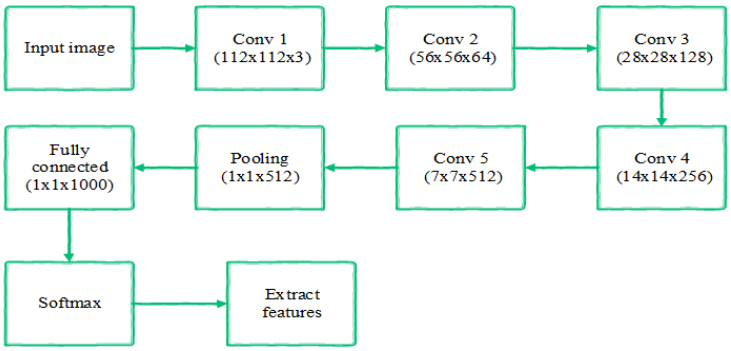

Figure 3. ResNet-18 model's architecture

The MinimumNorm approach puts spurious zeros within the unit circle and computes a desired noise subdomain vector based on the noise or information subspace eigenvectors to distinguish spurious zeros from true zeros. Therefore, the MinimumNorm approach employs a linear arrangement of all the noise subspace eigenvectors, whereas the Pisarenko technique makes use of the noise subspace eigenvector matching to the smallest eigenvalue. The following Eq. (25) Estimate of the Minimum-Norm PSD:

$$p_{\min}(f, n) = \frac{1}{|B(f)|^2} \tag{25}$$

where, n symbolizes the noise subspace's dimension.

4.5 Data Fusion

The extracted features are updated to the data fusion phase, for combining all features in a single vector form. The data fusion is performed based on two techniques as Concatenation-Based Fusion and Model-Based Fusion, which are detailed below.

Concatenation-based fusion: All the extracted features from the different features into a single feature vector. The combined feature vector $\overleftrightarrow{F}_v$ is represented in Eq. (26):

$$\overleftrightarrow{F}_v = \left[\overleftrightarrow{F}_s \left\| \overleftrightarrow{F}_i \right\| \overleftrightarrow{F}_t \left\| \overleftrightarrow{F}_f \right.\right] \tag{26}$$

Let, $\overleftrightarrow{F_S}$ is denoted as statistical features, $\overleftrightarrow{F}$ is denoted as image features, $\overleftrightarrow{F}$ is considered as time domain features, $\overleftrightarrow{F_f}$ is denoted as frequency domain features, and $\|$ is considered as concatenation operation.

Model-based fusion: The model-based fusion is performed using three neural network techniques for processing each feature of the input data. The statistical, image, and time and frequency domain features are intermediate with neural networks using Eqs. (27)-(29).

$$sta_{\text{int}} = NN\_statistical\,(Z_{sta}) \tag{27}$$

$$img_{\text{int}} = NN\_image\,(Z_{img}) \tag{28}$$

$$time\&freq_{\text{int}} = NN\_time\&frequencyl\left(Z_{time\&freq}\right) \tag{29}$$

Let, Z_{sta}, Z_{img}, and $Z_{time\&freq}$ are denoted as input features of different domains, and sta_{int}, img_{int}, and $time\&freq_{int}$ are the intermediate outputs of the different domain neural networks.

Then the intermediate outputs of the three models are concatenated to generate a single fused vector using Eq. (30).

$$\overleftrightarrow{F}_h = concatenate\left[sta_{\text{int}} img_{\text{int}} time\&freq_{\text{int}}\right] \tag{30}$$

Let, $\overleftrightarrow{F_h}$ is denoted as fused out of the three intermediate outputs of the neural network. Then the fused features are updated in the feature selection process for selecting best features to enhance prediction results.

4.6 Feature Selection

The best feature is selected with a fused dataset using slime mould optimization (SMO) [35], and Mothflame optimization (MFO) [36] called the HSMMF model.

Process of HSMMF

Larvae and adults are the two major life stages for moths. Within cocoons, the larvae transform into moths. The unique ways that moths navigate at night are the most fascinating aspect of their biology. They have learned to use the light of the full moon to fly at night. For navigation, they used a system known as transverse orientation. This technique uses a very efficient mechanism to keep the moth flying at a constant angle to the moon. The SMA actively seeks out a food source while taking on the form of the acellular Slime Mould (SM). Because of its unique anatomy, which enables it to create intricate venous networks connecting all of its food sources simultaneously, Slime Mould can accomplish this. Tubular veins circulate with the cytoplasm when Slime Moulds' biochemical oscillator detects a food source and sends contraction waves throughout the vein system. For the Slime Mould to locate food sources, it needs both positive and negative input. There are three stages to it: oscillation, wrap food, and approach food.

Approach Food

Based on the smell that food releases, Slime Mould may reach it. The contraction mode can be replicated using the following calculation methods, which can also be used to interpret the contraction phase's approaching behavior. A method for approaching food is found in Eq. (31).

$$\vec{Z}(t+1) = \begin{cases} \vec{Z}_{b(t)} + \vec{V}_b \times \left(\vec{w} \times \vec{Z}_{A(t)} - \vec{Z}_{B(t)}\right) & R < P \\ \vec{V}_c \times \vec{Z}(t) & R \geq P \end{cases} \tag{31}$$

Let, $\vec{V}_b$ be the variable that has possible values in the range of –1 (negative) and +1 (positive). Conversely, $\vec{V}_c$ declines linearly from 1 to $0, \vec{w}$ is the SM weight. t is denoted as the current iteration, $\vec{Z}_A$ and $\vec{Z}_B$ are the two individuals who were chosen at random via the SM, $\vec{Z}$ is considered a Slime Mould location, and $\vec{Z}_{b(t)}$ is the one place where there is now the highest concentration of scent. The formula for P is given by Eq. (32).

$$P = \tanh\left|s(i) - \vec{d}_f\right| \tag{32}$$

In Eq. (2), $\vec{d}_f$ represents the maximum degree of physical fitness reached throughout all iterations, $i \in 1, 2, \ldots, n, s(i)$ illustrates the state of $\vec{Z}$. The $\vec{V}_b$ utilizes Eqs. (33) and (34) to define it.

$$\vec{V}_b = [-k, k] \tag{33}$$

$$k = \arctan\left(-\left(\frac{t}{\max\_t}\right) + 1\right) \tag{34}$$

The $\vec{w}$ is measured using Eqs. (35) and (36).

$$\vec{w}(SI(i)) = \begin{cases} 1 + R.\log\left(\frac{b_f - s(i)}{b_f - \varpi_f} + 1\right) & condition \\ 1 - R.\log\left(\frac{b_f - s(i)}{b_f - \varpi_f} + 1\right) & others \end{cases} \tag{35}$$

$$SI(i) = sort(s) \tag{36}$$

Let, SI depicts the fitness rate arranged in ascending order, f symbolizes the llessfitness value currently attained during the iterative procedure, b_f shows the best fitness value attained during the current iteration, R signifies a

haphazard value within the range [0,1], the condition indicates that $s(i)$ in the upper echelon of the population, and max_t demonstrates the highest amount of iterations. The location of the individual being looked for $\vec{Z}$ perhaps changed by adjusting the settings $\vec{V}_b, \vec{V}_c, \vec{w}$ and It is possible to update the searcher's location to the most suitable location $\vec{Z}_{b(t)}$, that has just been acquired.

Wrap Food

This part simulates the device of venous tissue contraction used by SM during searching. More food intake causes the cytoplasm to flow more quickly, the bio-oscillator to produce a extra substantial wave, and the vein to become more noticeable. The situation that causes Slime Mould to alter its foraging tactics in answer to the quality of the food is mimicked. The weight in the location increases when there is sufficient food nearby; whenever there is not sufficient food nearby, the region loses relevance and the creature shifts its attention to other areas to investigate. In this stage, each moth's position about a flame is modified, exhibiting a spiral motion behavior. Eq. (37) is the calculation for the mathematical method utilized to update the SM's location.

$$\vec{Z}^* = \begin{cases} rand.(u_b - l_b) + l_b & rand < x \\ \vec{Z}_{b(t)} + \vec{V}_b \times \left(\vec{w} \times \vec{Z}_{A(t)} - \vec{Z}_{B(t)}\right) & R < P \\ \vec{V}_c.D_i.e^{qt}.\cos(2\pi t) + \vec{F}_j & R \geq P \end{cases} \tag{37}$$

Let, D_i be the distance of the i^{th} moth from the j^{th} flame, q be the constant, and t be the random number in [-1,1]. The bottom l_b and upper bounds u_b of the search range are in Eq. (37). Moreover, rand and R represent the value selected at random from the interval [0,1]. Finally, discuss the importance of the value throughout the experiment's parameter-modifying phase x.

Oscillation

The SM employs the propagation wave generated by the biological oscillator to alter the cytoplasmic movement in veins. It brings about a more favorable concentration of food for the Slime Mold. It's employed $\vec{V}_b, \vec{V}_c, \vec{w}$ to model the alterations that take place in Slime Mould by representing the changes in venous width. The production of the Slime Mold is the aim. to facilitate the slime mold's faster movement toward food once they find high-quality food, $\vec{w}$ Eq. (37) provides a mathematical description of the slime mold oscillation frequency, which is near to one for different feeding concentrations. The frequency at which food gets brought in is lower when there is less food concentrated in one area. It enhances Slime Mould's capacity to select the optimal food supply. The value increases with the number of iterations $\vec{V}_b$. Slime Mould's preferred behavior could be replicated by the cooperative relationship between $\vec{V}_b, \vec{V}_c$. Even when it has discovered a decent food source, Slime Mould will keep separating organic components in search of a better one. The purpose of this is to make Slime Mould more likely to come across a higher-quality food supply. Finally, the designed model chooses the greatest attributes from the dataset and enhances the prediction results of the developed model.

4.7 Graph-Logic Attention CNN-LSTM (GLACL-Net) Model Development

In this phase, a hybrid DL model is developed to enhance the prediction results of the lung-affected area and lung disease. The developed GLACL-Net model incorporates CNN-LSTM with attention and skip connection model, logic RNN, and graph neural network. The architecture of the developed model is shown in Figure 4.

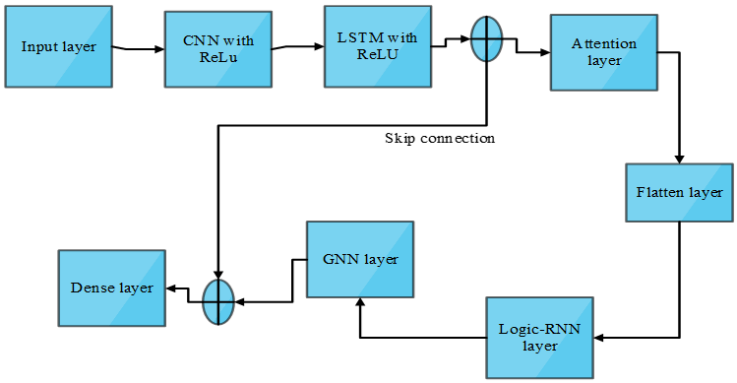

Figure 4. Architecture of the GLACL-Net model

CNN-LSTM [37]: To extract spatial features from image data using CNN and to capture temporal dependencies in sequential data using LSTM.

• Attention Mechanism: Incorporate Attention Layers to focus on the most relevant parts of the data also improves the model's ability to learn by selectively emphasizing important features.

• Skip Connections: to facilitate gradient flow and mitigate vanishing gradient problems. Improve training efficiency and model performance by allowing direct paths for gradient propagation.

• Graph Neural Network (GNN) [38]: used for complex structures and interactions in the data through graph-based features.

• Logic-RNN [39]: Enhance model interpretability and performance by incorporating logical rules and constraints.

Process of GLACL-Net model

The attention layer, LSTM system, and 1D CNN make up the three primary parts of the framework. 1D CNN uses feature extraction to extract information from the input data. This algorithm's CNN consists of several filter layers that organize throughout the input data to find pertinent features. A series of feature maps that are passed into the LSTM network comprise the CNN's output. The time-dependent relationships in the input data are captured by the LSTM network. The sequence of inputs of feature maps serves as the basis for training the multiple layers of LSTM cells that make up the LSTM network. A series of hidden states are produced by the LSTM network and supplied into the attention layer. The attention layer is used to intelligently focus on the relevant parts of the input pattern by weighing the importance of every hidden state in the output sequence. This increases the predicted accuracy of the system and helps it extract the most pertinent information given the input data. To do this, every hidden state's concentration score is first determined according to how relevant it is to the prediction job. The attention layer's output is then obtained by computing the weighted total of the value vectors using the weight vector. The attention layer can collect different facets of the input data since it might have many heads. Moreover, the skip incorporates the outputs of the Attention and LSTM layers to expand the model's capacity. It enables the model to take advantage of both layers' advantages and complementing qualities. The Attention layer concentrates on significant portions of the input sequence, while the LSTM layer records dependencies over time and encodes contextual data. Integrating their outputs can enhance the model's capacity to recognize intricate patterns and produce a fuller representation.

Let K be the input data with n models and m be the features and all sample is shown in an order of t timesteps. Let L be the target variable. Initially, modify the input data to take on the desired shape (n, m, t), where n is denoted as no. of samples, t is considered as no. of timesteps, and m is represented as no. of features. The mathematical expressions of each layer are described in Eqs. (38)-(45).

The input layer is obtained using Eq. (38).

$$i = K \in \Re^{n \times m \times t} \tag{38}$$

The CNN layer is obtained using Eq. (39).

$$p = cnn(K) \in \Re^{n \times m \times t \times p_{units}} \tag{39}$$

The LSTM layer is obtained using Eq. (40).

$$q = lstm(p) \in \Re^{n \times t \times q_{units}} \tag{40}$$

The Attention layer is obtained using Eq. (41).

$$A = attention(q) \in \Re^{n \times A_{units}} \tag{41}$$

The Skip Connection layer is obtained using Eq. (42).

$$s = q_{units} * c_{units} \tag{42}$$

The flattened layer is obtained using Eq. (43).

$$f = dense(s) \in \Re^{n \times t \times f_{wiits}} \tag{43}$$

To hide the redundant portion of the data, used a mask matrix while keeping in mind the feedback loop issue. The following changes were made to an RNN update by the designed framework using Eqs. (44) and (45).

$$g^t = f\left(ua^t + wg_c^{t-1} + y\right) \tag{44}$$

$$g_c^t = f\left(u'cem\left(a^t, mask\right) + w'g_c^{t-1} + y'\right) \tag{45}$$

Let, f is often a nonlinear activation function. $u\,(u')$, $w\,(w') \in \Re^{G \times d}$, and $y\,(y') \in \Re^G$ are denoted as trainable parameters. $g^t \in \Re^G$ is denoted as hidden layer output, and $g_c^{t-1} \in \Re^G$ is transferred to the following time step with the hidden layer vector. To incorporate the instance's logic rules, $a_j \rightarrow a_i$ identified as ε_k, a mix of $(\varepsilon_k, a_i, a_j)$

is acquired, and the contribution is $a^t = \varepsilon_k{}^t \oplus a_i{}^t \oplus a_j{}^t$, where $\oplus$ shows the operation of concatenation, and π is a virtual variable to keep things simple. The protective matrix, or "mask," is made up of the rule's relative weight and 0. The mask matrix has the same dimensions as a single logic rule. i.e. $d * 3$. In this case, the weight of the rule, represented by a non-zero cell, indicates the relative importance of the rules, while the cell with 0 is utilized to effectively eliminate redundant information. $cem(m1, m2)$ is a function that indicates that the two matrices have been increased by the matching elements and have the same dimensions. Here, G and d are the input's and the hidden layer's dimensionality.

There are two ways to create a temporal medical event graph: (1) time-aware graphs and (2) co-occurrence graphs. (3) a time-aware graph and a co-occurrence graph. The cooccurrence graph shows medical occurrences that occurred at that $i\left(z_i^{w_k}\right)$ are occasionally connected to medical actions $i+1\left(z_{i+1}^{w_k}\right)$. The graph is indicated by $x \in \sum \times l \times t(t = \{1, 2, \ldots, T\})$ and $x_i \in \sum \times l$ is the matrix of adjacency about $z_i^{w_k}$ and $z_{i+1}^{w_k}$. Upon acquiring the co-occurrence graph x, use GCN, a particular type of GNN, to show x as follows in Eqs. (46) and (47).

$$\hat{x} = \tilde{d}^{-\frac{1}{2}}(x + I)\tilde{d}^{-\frac{1}{2}} \tag{46}$$

$$y = mean\left\{\hat{x} relu\left(\hat{x} z m_1\right) m_2\right\} \tag{47}$$

Let, $\tilde{d}$ be the degree matrix that x, I be the identity matrix, w_k are denoted as weight matrices, and mean $\{.\}$ is denoted as a mean function. Create a time-aware graph by taking into account the varying time intervals between two neighboring time points x through swapping out $X_{jk} = 1 \in x_i$ by $\bar{X}_{jk} = \frac{1}{\Delta t_i} \in \bar{x}_i$, where Δt_i is the amount of time that passes between the i and time $i + 1$.

The output layer is obtained using Eq. (48).

$$L = dense(f) \in \Re^{n \times 1} \tag{48}$$

Let, cnn refers to the CNN layer that has p_{units} units, $lstm$ is regarded as an LSTM layer that q_{units} units, attention: Attention layer that generates an attention vector by using the LSTM layers as input A_{units} dimensions, dense is the FC layer with f_{units} units. The mean squared error loss function is the goal of the model's training using Eq. (49).

$$mse(L, l) = \frac{1}{n}\sum_{i=1}^{n}(L_i - l_i)^2 \tag{49}$$

Finally, the developed model predicts lung disease accurately with high performance. The developed model takes less execution time and less error rate to predict lung disease.

4.8 Decision Making

To clarify the model's predictions and shed light on how eye gazing and EHR data inform decision-making, incorporate XAI approaches (LIME, SHAP). This has the potential to increase acceptability and trust among medical practitioners.

Local Interpretable Model-Agnostics Explanations (LIME)

LIME is part of a larger class of removal-based explanations, which quantify the significance of features by statistically modeling their removal to prove feature importance [40]. LIME's primary features are its model-agnostic and local nature. A linear approximation near the choice point can provide a localized explanation, even though the global decision boundary may be convoluted and squiggly. Parts of the information required for the prediction are identified by the localized explanation, which adds to the explanations' intuitiveness and understandability. The majority of individuals would be satisfied with a straightforward, localized explanation, but complex edge cases, including uncommon medical disorders, require consideration in the global decision model. The decision-maker can also identify possible errors in the logic of the model by emphasizing the input parameters that influenced the choice.

The estimate for a certain instance a for an underlying system is represented in Eq. (50).

$$F_{model} : A \to B_{model}; a \in A \tag{50}$$

a: explained instance A: input feature space B_{model}: predicted target class for the underlying model LIME locally estimates choices made by $F_{model}(a)$ with $h_{lim}(a)$; The coefficient parameters c generated for $h_{lim}(a)$ represent the feature importance of the local model. LIME is defined as an optimization issue that weighs understanding and local fidelity loss with the objective function in Eq. (51).

$$\psi(a) = \arg\min l\left(F_{model}, h_{lime}, \Pi_a\right) + \Omega\left(h_{\text{lime}}\right) \tag{51}$$

where, $\Omega(h_{lime})$ is denoted as penalty function based on the difficulty of h_{lime}, l is considered as loss function, Π_a is denoted as weight distributed according to closeness to an instance a. The visualization graph using LIME is shown in Figure 5.

SHapley Additive exPlanations (SHAP)

A well-liked and useful method for figuring out how each feature affects a model's predictions is SHAP [41]. SHAP stands out because it offers information on both the direction and the extent of a feature's shift in a forecasted outcome. Whether variations in patients' degrees of contact with their medical providers would be skewing model output after using SHAP to identify a model's top predictors.

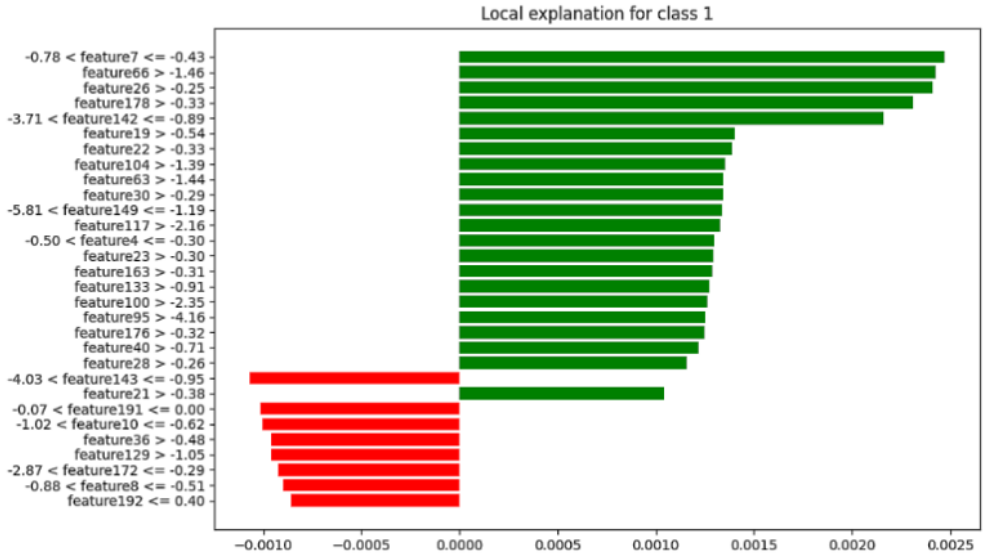

Figure 5. LIME tool-based visualization

In the MIMIC-Eye framework, SHAP can be employed to provide transparent and interpretable insights into how EHR and eye gaze data influence clinical DSS outcomes. The cooperative the theory of games provides the basis for the Shapley numbers. They give each feature a fair value that reflects how much it contributes to the algorithm's prediction. For a prediction model p, the Shapley value for a feature i in Eq. (52).

$$\varphi_i = \sum_{\bar{s} \subseteq \bar{f}/\{i\}} \frac{|\bar{s}|!(|\bar{f}| - |\bar{s}| - 1)!}{|\bar{f}|!} (\bar{f}(\bar{s} \cup \{i\}) - \bar{f}(\bar{s})) \tag{52}$$

Let, $\bar{s}$ is denoted as a features subset, $\bar{f}$ is considered a complete feature set, and $\bar{f}(\bar{s})$ is the model's forecast using the characteristics in $\bar{s}$.

Last but not least, LIME SHAP values are useful applications of Shapley values that break down the forecasting ability of a model into the effects of each feature. To determine how each feature in the EHR and eye gazing datasets contributes to the algorithm's predictions, compute its SHAP value. Analyze SHAP values for various predictions to see how certain features affect a given choice. This helps give thorough justifications for particular patient situations. The visualization graph using SHAP is shown in Figure 6.

5 Results and Discussion

The created model's output is integrated into the Python program, and the technique's performance metrics are verified against other DL approaches already in use. Through the application of EAI approaches to visualize the prediction outcomes, the multi-modal dataset is utilized to discover abnormalities and improve DSS.

5.1 Performance Analysis

Accuracy, precision, false positive rate (FPR), specificity, sensitivity, Matthew's correlation coefficient (MCC), F-measure, negative prediction value (NPV), false negative rate (FNR), and other performance metrics were utilized to validate the efficacy of the constructed model. The methods that are now in use versus the developed method are YOLOv5 [42], RNN [43], VGG-NET [44], and 3D-CNN [45]. The comparative findings between the developed and current procedures are displayed in Table 2.

Performance metrics for several DL models, including 3DCNN, RNN, VGGNET, YOLOv5, and a suggested model, are shown in Table 2. The suggested model achieves the highest accuracy, precision, sensitivity, specificity,

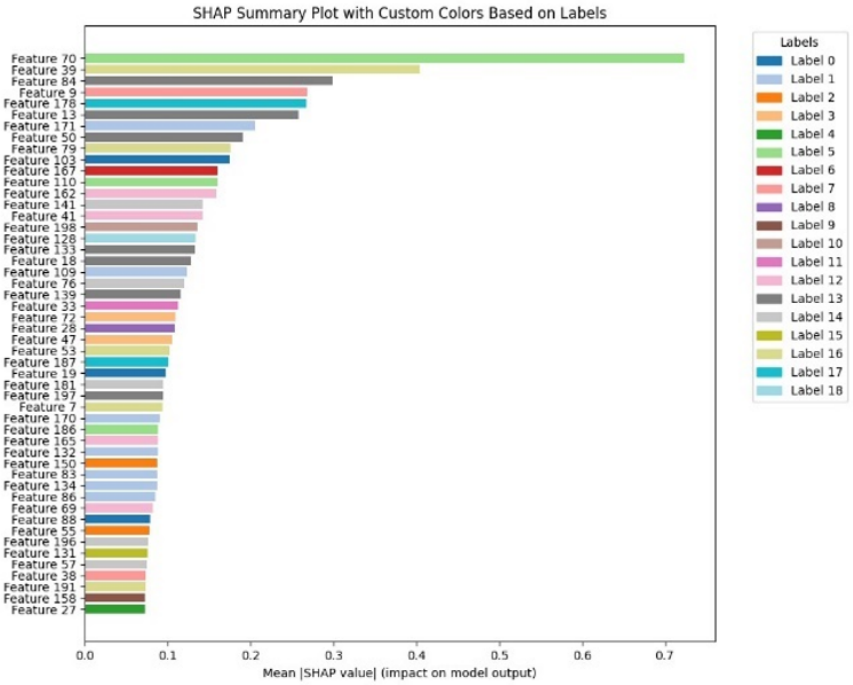

Figure 6. SHAP tool-based visualization

F-measure, MCC, and NPV overall, outperforming the other models on the majority of criteria. Additionally, it shows the lowest FPR and FNR, demonstrating superior ability in accurately classifying situations as positive and negative. The comparative findings between the developed and current procedures are displayed in Table 3.

Table 2. Comparison results of the developed model with existing techniques for 70 learning rates

Performance Metrics	3DCNN	RNN	VGGNET	YOLOv5	Proposed
Accuracy	0.8174	0.9311	0.8416	0.9426	0.9721
Precision	0.8367	0.9062	0.83	0.9295	0.9535
Sensitivity	0.7736	0.8542	0.8469	0.9292	0.988
Specificity	0.8584	0.9162	0.8365	0.9457	0.9783
F-Measure	0.8039	0.9111	0.8384	0.9243	0.9704
MCC	0.6352	0.8304	0.6832	0.9052	0.9645
NPV	0.8017	0.8542	0.8529	0.916	0.9892
FPR	0.0916	0.0838	0.1035	0.0643	0.0417
FNR	0.1064	0.0958	0.1131	0.0708	0.012
Scalability					

Table 3. Comparison results of the developed model with existing techniques for 80 learning rates

Performance Metrics	3DCNN	RNN	VGGNET	YOLOv5	Proposed
Accuracy	0.8507	0.9447	0.899	0.955	0.9841
Precision	0.9263	0.9204	0.9184	0.9312	0.9899
Sensitivity	0.7928	0.9327	0.8738	0.9096	0.98
Specificity	0.9222	0.9379	0.9238	0.9295	0.9888
F-Measure	0.8544	0.9263	0.8955	0.904	0.9849
MCC	0.7122	0.8898	0.7989	0.9035	0.9682
NPV	0.783	0.9286	0.8818	0.9168	0.9778
FPR	0.0778	0.0721	0.0762	0.0805	0.0112
FNR	0.1172	0.0673	0.1262	0.0904	0.02

The performance metrics of several DL models, including 3DCNN, RNN, VGGNET, YOLOv5, and a suggested

model, are compiled in Table 3. With the best accuracy (0.9841), precision (0.9899), sensitivity (0.98), specificity (0.9888), F-measure (0.9849), MCC (0.9682), and NPV (0.9778), the suggested model outperforms the competition in practically every metric. Additionally, it registers the lowest FNR (0.02) and FPR (0.0112). Additionally, Figures 7– 15 compare the suggested model's performance metrics with those of the current models, which have two learning rates of 70 and 80.

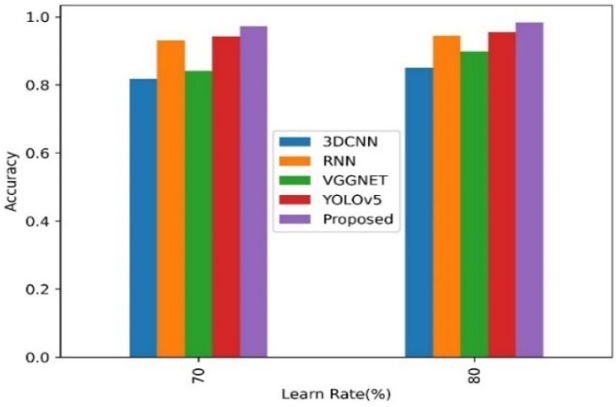

Figure 7. Accuracy comparison

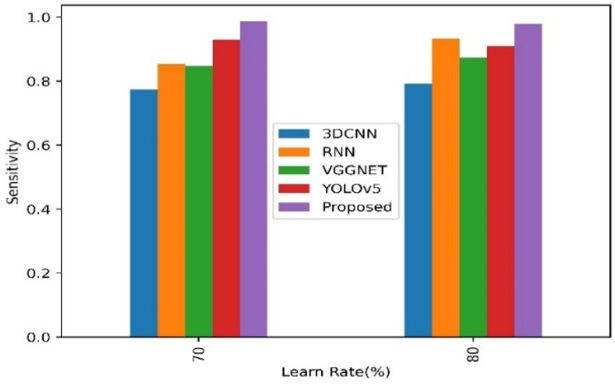

Figure 8. Sensitivity comparison

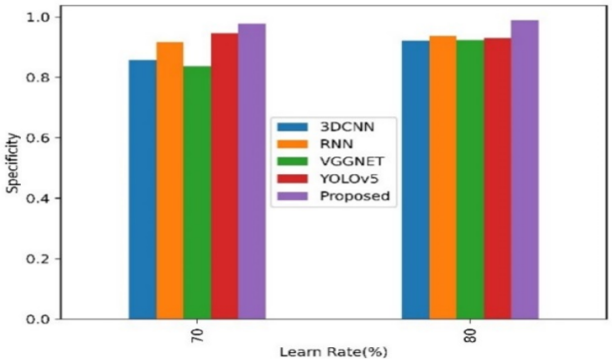

Figure 9. Specificity comparison

For 70 learning rates, the suggested model has the highest accuracy (0.9721), followed by YOLOv5 (0.9426), RNN (0.9311), VGGNet (0.8416), and 3DCNN (0.8174). For 80 learning rates, the suggested model retains the highest accuracy (0.9841) when compared to YOLOv5 (0.955), RNN (0.9447), VGGNet (0.899), and 3DCNN (0.8507).

With a sensitivity of 0.988, the suggested model outperforms RNN (0.8542) and YOLOv5 (0.9292). At a 70-learning rate, VGGNet trails at 0.8469, and 3DCNN trails at 0.7736. With an 80-learning rate, the suggested

model continues to lead in sensitivity (0.98), followed by RNN (0.9327), YOLOv5 (0.9096), VGGNet (0.8738), and 3DCNN (0.7928).

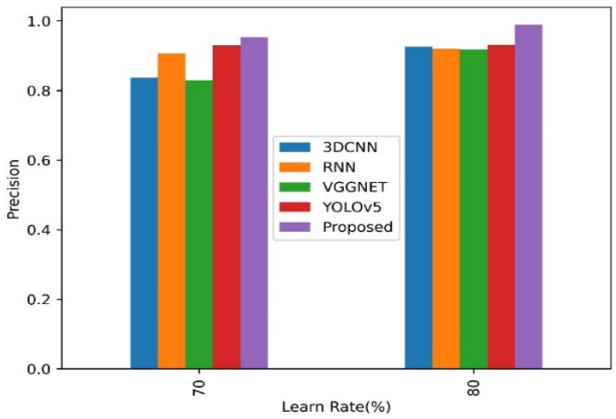

Figure 10. Precision comparison

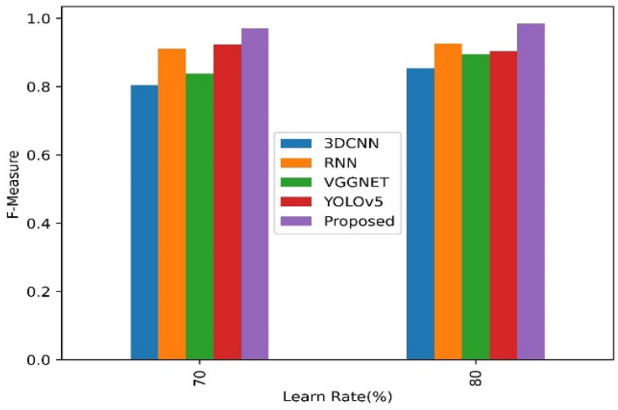

Figure 11. F-measure comparison

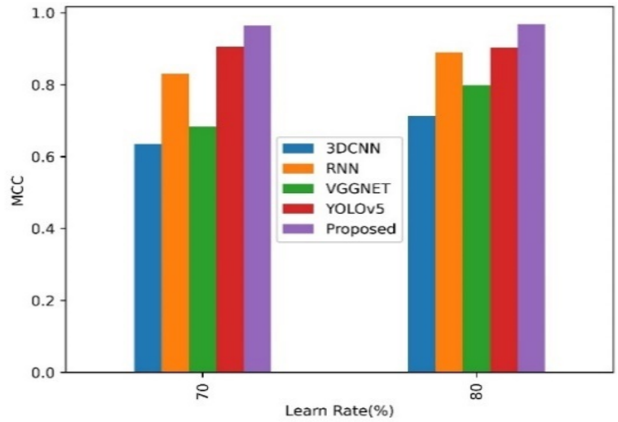

Figure 12. MCC comparison

The best-performing models are the suggested model (0.9783), followed by YOLOv5 (0.9457), RNN (0.9162), and 3DCNN (0.8584). At 0.8365, VGGNet has the lowest score for a 70-learning rate. The maximum specificity (0.9888) is found in the proposed model, with VGGNet (0.9238) and RNN (0.9379) following closely behind. For an 80-learning rate, YOLOv5 (0.9295) and 3DCNN (0.9222) are used and shown in Figure 9.

Additionally, the precision of the suggested model is excellent (0.9535), followed by YOLOv5 (0.9295) and RNN (0.9062) shown in Figure 10. For 70 learning rates, VGGNet (0.83) and 3DCNN (0.8367) exhibit poorer precision.

The suggested model continues to have better precision (0.9899) at this learning rate of 80, followed by VGGNet (0.9184), RNN (0.9204), and 3DCNN (0.9263).

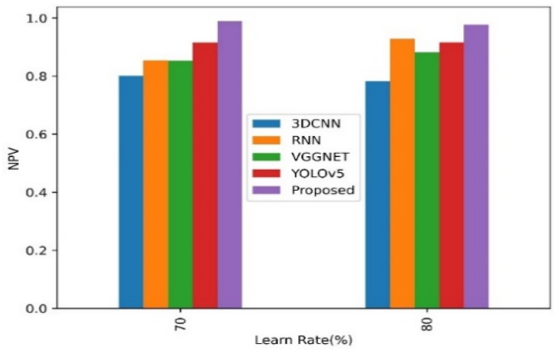

Figure 13. NPV comparison

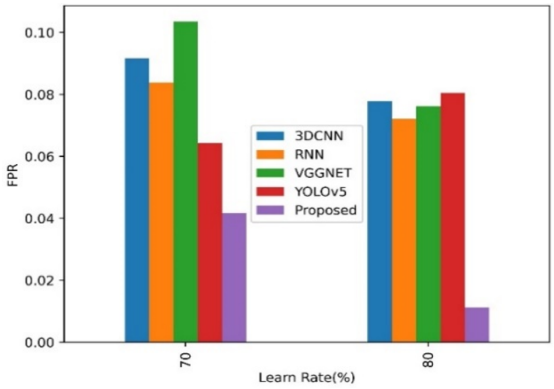

Figure 14. FPR comparison

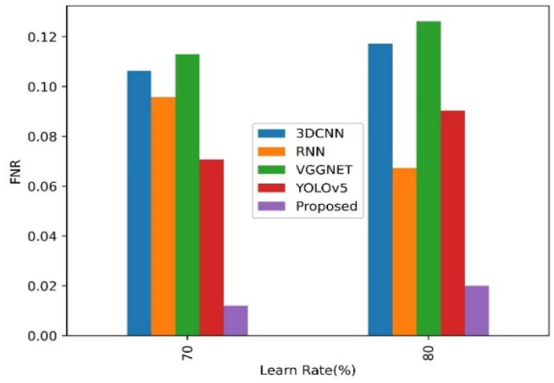

Figure 15. FNR comparison

The maximum F-Measure (0.9704) is achieved by the proposed model; for 70 learning rates, YOLOv5 (0.9243), RNN (0.9111), VGGNet (0.8384), and 3DCNN (0.8039) score lower. For 80 learning rates, VGGNet and 3DCNN score 0.8955 and 0.8544, respectively. The suggested approach achieves the maximum F-Measure (0.9849). Figure 11 shows the F-measure comparison.

The suggested model has the greatest MCC (0.9645), which is followed by RNN (0.8304) and YOLOv5 (0.9052). For 70 learning rates, the MCCs of VGGNet and 3DCNN are lower, at 0.6832 and 0.6352. Once again, the suggested model (0.9682) performs best in MCC, followed by RNN (0.8898) and YOLOv5 (0.9035). The MCC values of VGGNet (0.7989) and 3DCNN (0.7122) are lower for learning rates of 80. MCC comparison are shown in Figure 12.

With a 70-learning rate, the suggested model (0.9892) outperforms the others in NPV, including YOLOv5 (0.916), VGGNet (0.8529), RNN (0.8542), and 3DCNN (0.8017). With RNN (0.9286), YOLOv5 (0.9168), VGGNet

(0.8818), and 3DCNN (0.783) at an 80-learning rate, the suggested model has the highest NPV (0.9778). Figure 13 shows the NPV comparison.

For 70 learning rates, the suggested model outperforms YOLOv5 (0.0643), RNN (0.0838), 3DCNN (0.0916), and VGGNet (0.1035) in terms of false positive rate (FPR; 0.0417). For 80 learning rates, the suggested model has the lowest FPR (0.0112) when compared to VGGNet (0.0762), RNN (0.0721), YOLOv5 (0.0805), and 3DCNN (0.0778). Figure 14 shows the FPR comparison.

FNR comparison shows in Figure 15. With scores higher for 70 learning rates than YOLOv5 (0.0708), RNN (0.0958), VGGNet (0.1131), and 3DCNN (0.1064), the suggested model also has the lowest FNR (0.012), indicating fewer false negatives. When considering 80 learning rates, RNN (0.0673), YOLOv5 (0.0904), VGGNet (0.1262), and 3DCNN (0.1172) scored more than the suggested model, which has the lowest FNR (0.02), suggesting superior performance.

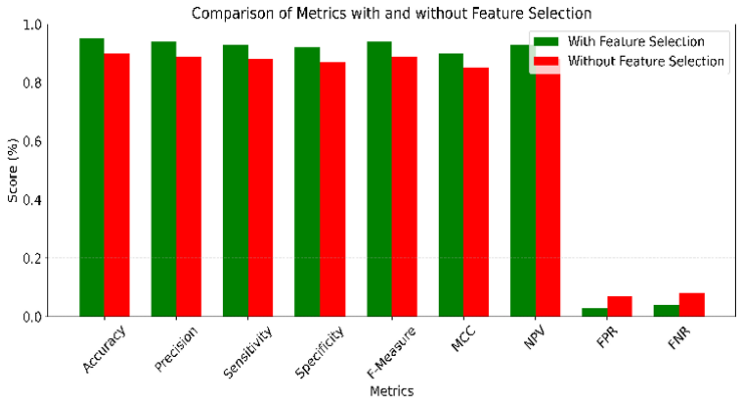

Figure 16. Developed model performance with and without feature selection

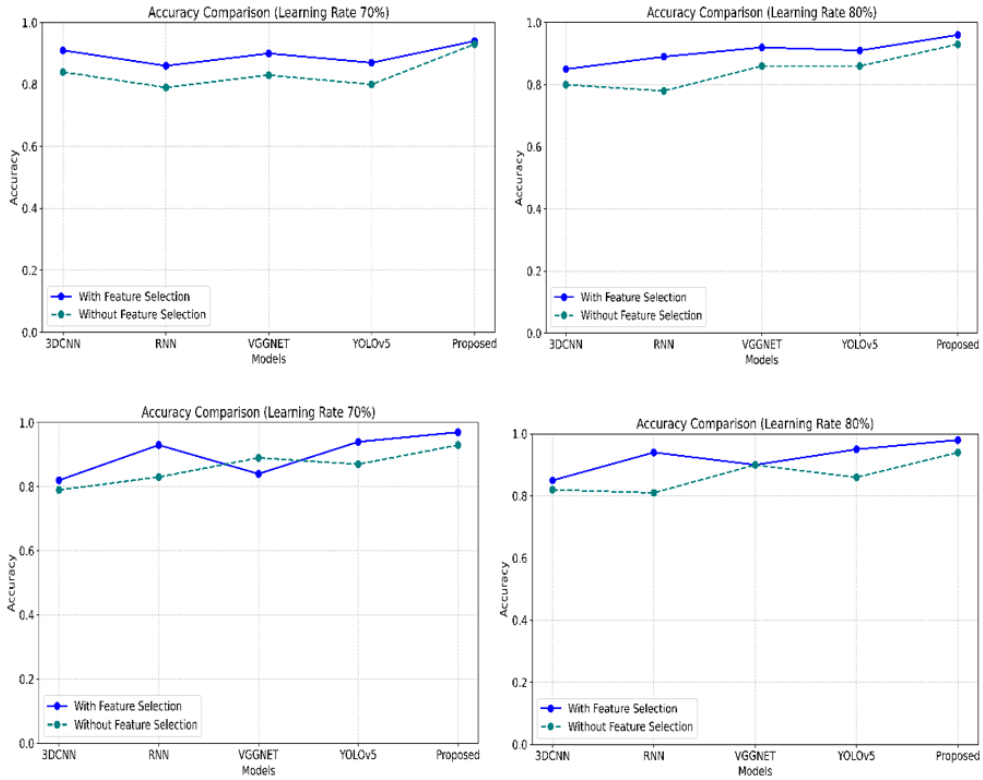

Figure 17. Accuracy comparison with concatenate fusion and model-based fusion

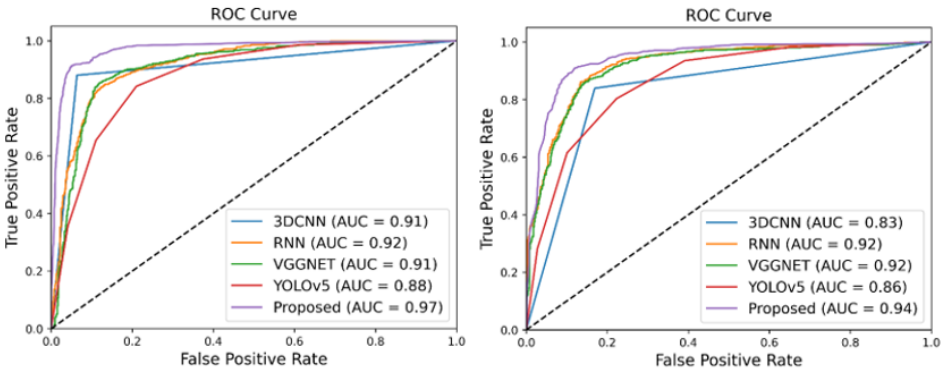

Figure 18. ROC curve comparison with concatenate fusion and model-based fusion (a) Concatenate fusion (b) Model-based fusion

5.2 Discussion

Figure 16 illustrates the comparison of various performance metrics for a developed technique model with and without feature selection. The green bars represent the scores achieved with feature selection, while the red bars represent the scores without feature selection. From the graph, it is evident that the proposed model performs better across most metrics when feature selection is applied. This indicates that feature selection enhances the model's ability to correctly identify relevant patterns in the data, leading to improved overall performance. Conversely, the metrics FPR and FNR, which ideally should be lower, also reflect the benefits of feature selection.

Overall, the graph demonstrates that incorporating feature selection into the model training process results in more accurate and reliable predictions, highlighting the importance of selecting the most relevant features for optimal model performance. Also, the performance of the feature fusion with concatenate fusion and modal-based fusion is validated with accuracy, which is shown in Figure 17.

The modal-based fusion gained higher performance while comparing concatenate fusion models. It shows the effectiveness of the developed model to predict the disease. The ROC curve of concatenate fusion and modal-based fusion is shown in Figure 18.

6 Conclusion

To improve radiologists' choices, we tackle the problem of integrating numerous data modalities in medical imaging in this study. Through the use of an extensive dataset that includes ECG, NIH chest X-ray, and MIMIC-III EHR data, we go beyond the conventional image-only method in the realm of medical DL applications. The proposed methodology uses cutting-edge techniques like min-max scaling and MICE for efficient preprocessing of EHR data. HMMs are utilized for segmentation and Deep Kalman Filters and Denoising Autoencoders are used for denoising to enhance the quality of image data. Concatenation and modal-based feature fusion are utilized to ensure secure data integration by merging features from chest X-ray pictures, ECG, and EHR without disclosing the raw data. The system's ability to assess the significance of every data modality for certain prediction tasks is made possible by the incorporation of an attention mechanism grounded in Ensemble DL models. This improves the interpretability and performance of the model. Moreover, transparency in model predictions is achieved through LIME and SHAP techniques, which offer insights into the impact of different data features on the model's decisions. The gained experimental outcomes are validated with existing DL techniques and gained impressive accuracy of 0.9841, precision of 0.9899, sensitivity of 0.98, NPV of 0.9778, less FPR of 0.0112, and FNR of 0.02.

Data Availability

All the data is collected from the simulation reports of the software and tools used by the authors. Authors are working on implementing the same using real world data with appropriate permissions.

Conflicts of Interest

The authors declare that they have no conflict of interest.

References

[1] M. Siddiq, "Ml-based medical image analysis for anomaly detection in CT scans, X-rays, and MRIs," *Devotion: J. Res. Community Serv.*, vol. 2, no. 1, pp. 53–64, 2020. https://doi.org/10.59188/devotion.v3i13.469

[2] S. Hussain, I. Mubeen, N. Ullah, S. S. U. D. Shah, B. A. Khan, M. Zahoor, R. Ullah, F. A. Khan, and M. A. Sultan, "Modern diagnostic imaging technique applications and risk factors in the medical field: A review," *BioMed Res. Int.*, vol. 2022, no. 1, p. 5164970, 2022. https://doi.org/10.1155/2022/5164970

[3] A. Koul, R. K. Bawa, and Y. Kumar, "Artificial intelligence in medical image processing for airway diseases," in *Connected e-Health: Integrated IoT and Cloud Computing*. Springer, 2022, pp. 217–254. https://doi.org/10.1007/978-3-030-97929-4_10

[4] J. Hofmeister, N. Garin, X. Montet, M. Scheffler, A. Platon, P. Poletti, J. Stirnemann, M. Debray, Y. Claessens, X. Duval, and V. Prendki, "Validating the accuracy of deep learning for the diagnosis of pneumonia on chest x-ray against a robust multimodal reference diagnosis: A post hoc analysis of two prospective studies," *Eur. Radiol. Exp.*, vol. 8, no. 1, p. 20, 2024. https://doi.org/10.1186/s41747-023-00416-y

[5] M. Arslan, A. Haider, M. Khurshid, S. S. U. A. Bakar, R. Jani, F. Masood, T. Tahir, K. Mitchell, S. Panchagnula, and K. M. Mitchell, "From pixels to pathology: Employing computer vision to decode chest diseases in medical images," *Cureus*, vol. 15, no. 9, p. e45587, 2023. https://doi.org/10.7759/cureus.45587

[6] B. Abhisheka, S. K. Biswas, B. Purkayastha, D. Das, and A. Escargueil, "Recent trend in medical imaging modalities and their applications in disease diagnosis: A review," *Multimed. Tools Appl.*, vol. 83, no. 14, pp. 43 035–43 070, 2024. https://doi.org/10.1007/s11042-023-17326-1

[7] H. Malik, M. S. Farooq, A. Khelifi, A. Abid, J. N. Qureshi, and M. Hussain, "A comparison of transfer learning performance versus health experts in disease diagnosis from medical imaging," *IEEE Access*, vol. 8, pp. 139 367–139 386, 2020. https://doi.org/10.1109/ACCESS.2020.3004766

[8] M. Rana and M. Bhushan, "Machine learning and deep learning approach for medical image analysis: Diagnosis to detection," *Multimed. Tools Appl.*, vol. 82, no. 17, pp. 26 731–26 769, 2023. https://doi.org/10.1007/s11042-022-14305-w

[9] S. M. Shafi, "An analysis of deep learning in CXR medical image processing," *J. Pharm. Negat. Result.*, vol. 13, pp. 701–709, 2022.

[10] S. Kumar and H. Kumar, "Lungcov: A diagnostic framework using machine learning and imaging modality," *Int. J. Tech. Phys. Prob. Eng.*, vol. 14, no. 51, p. 2, 2022.

[11] Y. A. Kadhim, M. U. Khan, and A. Mishra, "Deep learning-based computer-aided diagnosis (CAD): Applications for medical image datasets," *Sensors*, vol. 22, no. 22, p. 8999, 2022. https://doi.org/10.3390/s22228999

[12] J. Amann, A. Blasimme, E. Vayena, D. Frey, V. I. Madai, and P. Consortium, "Explainability for artificial intelligence in healthcare: A multidisciplinary perspective," *BMC Med. Inform. Decis. Making*, vol. 20, no. 1, p. 310, 2020. https://doi.org/10.1186/s12911-020-01332-6

[13] K. Govindan, H. Mina, and B. Alavi, "A decision support system for demand management in healthcare supply chains considering the epidemic outbreaks: A case study of coronavirus disease 2019 (COVID-19)," *Transport. Res. Part E: Logist. Transport. Rev.*, vol. 138, p. 101967, 2020. https://doi.org/10.1016/j.tre.2020.101967

[14] M. Javaid, A. Haleema, and R. P. Singh, "ChatGPT for healthcare services: An emerging stage for an innovative perspective," *BenchCouncil Trans. Benchmarks, Stand. Evaluations*, vol. 3, no. 1, p. 100105, 2023. https://doi.org/10.1016/j.tbench.2023.100105

[15] S. Rani, M. Chauhan, A. Kataria, and A. Khang, "IoT equipped intelligent distributed framework for smart healthcare systems," in *Towards the Integration of IoT, Cloud and Big Data: Services, Applications and Standards*. Springer Nature Singapore, 2023, pp. 97–114. https://doi.org/10.1007/978-981-99-6034-7_6

[16] A. Khatoon, "A blockchain-based smart contract system for healthcare management," *Electronics*, vol. 9, no. 1, p. 94, 2020. https://doi.org/10.3390/electronics9010094

[17] M. S. Hossain, G. Muhammad, and N. Guizani, "Explainable AI and mass surveillance system-based healthcare framework to combat COVID-I9 like pandemics," *IEEE Network*, vol. 34, no. 4, pp. 126–132, 2020. https://doi.org/10.1109/MNET.011.2000458

[18] C. Ma, H. Jiang, W. Chen, Z. Wu, X. Yu, F. Zeng, L. Guo, D. Zhu, T. Zhang, D. Shen, T. Liu, and X. Li, "Eye-gaze guided multi-modal alignment framework for radiology," *arXiv Preprint*, vol. arXiv:2403.12416, 2024. https://doi.org/10.48550/arXiv.2403.12416

[19] C. Hsieh, C. Ouyang, J. C. Nascimento, J. Pereira, J. Jorge, and C. Moreira, "Mimic-eye: Integrating mimic datasets with reflacx and eye gaze for multimodal deep learning applications," *PhysioNet*, vol. Version 1.0.0, 2023. https://doi.org/10.13026/pc72-as03

[20] Y. Kim, J. Wu, Y. Abdulle, Y. Gao, and H. Wu, "Enhancing human-computer interaction in chest x-ray analysis using vision and language model with eye gaze patterns," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer Nature Switzerland, 2024, pp. 184–194. https://doi.org/10.1007/978-3-031-72384-1_18

[21] C. Hsieh, A. Luís, J. Neves, I. B. Nobre, S. C. Sousa, C. Ouyang, J. Jorge, and C. Moreira, "Eyexnet: Enhancing abnormality detection and diagnosis via eye-tracking and x-ray fusion," *Mach. Learn. Knowl. Extr.*, vol. 6, no. 2,

pp. 1055–1071, 2024. https://doi.org/10.3390/make6020048

[22] C. Hsieh, I. B. Nobre, S. C. Sousa, C. Ouyang, M. Brereton, J. C. Nascimento, J. Jorge, and C. Moreira, "Mdf-net for abnormality detection by fusing x-rays with clinical data," *Sci. Rep.*, vol. 13, no. 1, p. 15873, 2023. https://doi.org/10.1038/s41598-023-41463-0

[23] Q. Yin, L. Zhong, Y. Song, L. Bai, Z. Wang, C. Li, Y. Xu, and X. Yang, "A decision support system in precision medicine: Contrastive multimodal learning for patient stratification," *Ann. Oper. Res.*, vol. 348, no. 1, pp. 579–607, 2025. https://doi.org/10.1007/s10479-023-05545-6

[24] M. I. Sharif, M. A. Khan, M. Alhussein, K. Aurangzeb, and M. Raza, "A decision support system for multimodal brain tumor classification using deep learning," *Complex Intell. Syst.*, vol. 8, no. 4, pp. 3007–3020, 2022. https://doi.org/10.1007/s40747-021-00321-0

[25] N. Gaw, T. J. Schwedt, C. D. Chong, T. Wu, and J. Li, "A clinical decision support system using multi-modality imaging data for disease diagnosis," *IISE Trans. Healthcare Syst. Eng.*, vol. 8, no. 1, pp. 36–46, 2018. https://doi.org/10.1080/24725579.2017.1403520

[26] G. Joshi, R. Walambe, and K. Kotecha, "A review on explainability in multimodal deep neural nets," *IEEE Access*, vol. 9, pp. 59 800–59 821, 2021. https://doi.org/10.1109/ACCESS.2021.3070212

[27] K. M. Boehm, P. Khosravi, R. Vanguri, J. Gao, and S. P. Shah, "Harnessing multimodal data integration to advance precision oncology," *Nat. Rev. Cancer*, vol. 22, no. 2, pp. 114–126, 2022. https://doi.org/10.1038/s41568-021-00408-3

[28] "In Hospital Mortality Prediction," 2023. https://www.kaggle.com/datasets/muhammaddzakykhairy/in-hospital-mortality-prediction-subset-and-filled

[29] "NIH Chest X-rays," 2023. https://www.kaggle.com/datasets/nih-chest-xrays/data

[30] "Ecg heartbeat categorization dataset," 2023. https://www.kaggle.com/datasets/shayanfazeli/heartbeat

[31] Z. Wang, O. Akande, J. Poulos, and F. Li, "Are deep learning models superior for missing data imputation in large surveys? Evidence from an empirical comparison," *arXiv Preprint*, vol. arXiv:2103.09316, 2021. https://doi.org/10.48550/arXiv.2103.09316

[32] P. Russo, F. Di Ciaccio, and S. Troisi, "Danae: A denoising autoencoder for underwater attitude estimation," *arXiv Preprint*, vol. arXiv:2011.06853, 2020. https://doi.org/10.48550/arXiv.2011.06853

[33] N. Roth, A. Küderle, M. Ullrich, T. Gladow, F. Marxreiter, J. Klucken, B. M. Eskofier, and F. Kluge, "Hidden Markov Model based stride segmentation on unsupervised free-living gait data in Parkinson's disease patients," *J. Neuroeng. Rehabil.*, vol. 18, no. 1, p. 93, 2021. https://doi.org/10.1186/s12984-021-00883-7

[34] I. A. Ahmed, E. M. Senan, T. H. Rassem, M. A. Ali, H. S. A. Shatnawi, S. M. Alwazer, and M. Alshahrani, "Eye tracking-based diagnosis and early detection of autism spectrum disorder using machine learning and deep learning techniques," *Electronics*, vol. 11, no. 4, p. 530, 2022. https://doi.org/10.3390/electronics11040530

[35] S. Li, H. Chen, M. Wang, A. A. Heidari, and S. Mirjalili, "Slime mould algorithm: A new method for stochastic optimization," *Future Gener. Comp. Syst.*, vol. 111, pp. 300–323, 2020. https://doi.org/10.1016/j.future.2020.03.055

[36] S. Mirjalili, "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm," *Knowl.-Based Syst.*, vol. 89, pp. 228–249, 2015. https://doi.org/10.1016/j.knosys.2015.07.006

[37] N. V. Dharwadkar, V. H. Kalmani, and V. Thapa, "Crop yield prediction using deep learning algorithm based on CNN-LSTM with Attention Layer and Skip Connection," *Research Square*, 2023. https://doi.org/10.21203/rs.3.rs-3118781/v1

[38] S. Liu, T. Li, H. Ding, B. Tang, X. Wang, Q. Chen, J. Yan, and Y. Zhou, "A hybrid method of recurrent neural network and graph neural network for next-period prescription prediction," *Int. J. Mach. Learn. Cybern.*, vol. 11, no. 12, pp. 2849–2856, 2020. https://doi.org/10.1007/s13042-020-01155-x

[39] B. Chen, Z. Hao, X. Cai, R. Cai, W. Wen, J. Zhu, and G. Xie, "Embedding logic rules into recurrent neural networks," *IEEE Access*, vol. 7, pp. 14 938–14 946, 2019. https://doi.org/10.1109/ACCESS.2019.2892140

[40] Y. Hailemariam, A. Yazdinejad, R. M. Parizi, G. Srivastava, and A. Dehghantanha, "An empirical evaluation of AI deep explainable tools," in *2020 IEEE Globecom Workshops (GC Wkshps)*, 2020, pp. 1–6. https://doi.org/10.1109/GCWkshps50303.2020.9367541

[41] V. Vimbi, N. Shaffi, and M. Mahmud, "Interpreting artificial intelligence models: A systematic review on the application of LIME and SHAP in Alzheimer's disease detection," *Brain Inform.*, vol. 11, no. 1, p. 10, 2024. https://doi.org/10.1186/s40708-024-00222-1

[42] D. Kvak, A. Chromcová, R. Hrubý, E. Janů, M. Biroš, M. Pajdaković, K. Kvaková, M. A. Al-antari, P. Polášková, and S. Strukov, "Leveraging deep learning decision-support system in specialized oncology center: A multi-reader retrospective study on detection of pulmonary lesions in chest x-ray images," *Diagnostics*, vol. 13, no. 6, p. 1043, 2023. https://doi.org/10.3390/diagnostics13061043

[43] K. Shankar, E. Perumal, V. G. Díaz, P. Tiwari, D. Gupta, A. K. J. Saudagar, and K. Muhammad, "An optimal cascaded recurrent neural network for intelligent COVID-19 detection using chest X-ray images," *Appl. Soft Comput.*, vol. 113, p. 107878, 2021. https://doi.org/10.1016/j.asoc.2021.107878

[44] M. Qjidaa, A. Ben-Fares, Y. Mechbal, H. Amakdouf, M. Maaroufi, B. Alami, and H. Qjidaa, "Development of a clinical decision support system for the early detection of COVID-19 using deep learning based on chest radiographic images," in *2020 International Conference on Intelligent Systems and Computer Vision (ISCV)*. IEEE, 2020, pp. 1–6. https://doi.org/10.1109/ISCV49265.2020.9204282

[45] A. Masood, P. Yang, B. Sheng, H. Li, P. Li, J. Qin, V. Lanfranchi, J. Kim, and D. D. Feng, "Cloud-based automated clinical decision support system for detection and diagnosis of lung cancer in chest CT," *IEEE J. Transl. Eng. Health Med.*, vol. 8, pp. 1–13, 2019. https://doi.org/10.1109/JTEHM.2019.2955458