ISSN: 2096-3246
Volume 58, Issue 01, 2026

A HYBRID CNN–LSTM WITH XAI FOR HEART-DISEASE PREDICTION USING MULTI-DATASET INTEGRATION

R Kamal Krishna[1], S. Gopinathan[2]
[1]Ph.D Research Scholar, Department of Computer Science, University of Madras, India, Tamilnadu
[2]Professor, Department of Computer Science, University of Madras, India, Tamilnadu

ABSTRACT
This precision in detecting cardiac risk at an early phase is still one of the most important requirements in the clinical decision support. The present paper suggests the development of a comprehensive analytical structure of prediction of heart disease with the help of structured clinical features on the UCI Cleveland dataset. During data preprocessing, feature transformation, feature normalization, and feature encoding as well as dimensionality reduction of features is performed to apply machine learning and deep learning. The comparison of different machine learning classifiers is conducted, and a hybrid deep learning system of CNN and LSTM is created to represent the local dependencies and longitudinal relationships that are inherent to the features of the patient. A comparative evaluation using the traditional machine learning algorithms and deep learning models validates enhanced predictive stability with enhanced generalization results of the hybrid model. It highlights the suitability of the integrated deep learning arrangements in terms of medical risk forecasting using clinically heterogeneous indicators.
Keywords: Heart Disease Prediction, Deep Learning, CNN–LSTM Hybrid Architecture, Clinical Data Analytics, Medical Decision Support, Machine Learning.

1- Introduction
Heart disease is one of the health burdens in the world, as it is estimated that 17.9 million people die every year. Early diagnosis consequently enables early medical treatment. Clinical acumen, ECG interpretation, and laboratory analysis have always been very relied in the traditional diagnosis of heart conditions. Against the background of a swift increase in the amount of digital healthcare data, the use of ML and DL methods is becoming more widespread to enhance risk stratification.
All the past literature is performed on single datasets, which are usually small and lack diversity and limit their generalizability. Further, a lot of deep learning models are black boxes, and they cannot be interpreted, a significant consideration when implementing them in clinical use. It is against this backdrop that the present work introduces a Hybrid CNN-LSTM model that is trained on various datasets, SHAP explainability, and ensemble learning, as well as statistical validation. The application of machine-learning to predict heart-disease has also cut across the classical machine-learning pipelines, more recent deep-learning models based on both structured and imaging data, combining multiple datasets, and new explainable-AI models. The fifteen original and timely works [1-15] used in the review below are indicative of the strides towards those directions and gaps that the proposed study fills.
The earlier prediction algorithms have been applied to the structured UCI data and most of them are

classical machine-learning algorithms. In [11], the author uses the Decision Trees, Naive Bayes and Logistic Regression and SVM to the Cleveland data and gives results of the highest accuracy of approximately 85 percent and reflects the impact of simple feature-selection and evaluation scores. In the same vein, comparing [12] of the Logistic Regression, KNN, SVM and ANN, one may see that KNN has been doing as good as it could do in comparison with other models, though the preprocessing is done right, and the importance of risk-factors is considered appropriately. The larger set of algorithmic comparisons used in [13] takes advantage of greater number of algorithmic comparisons used in the classification of highest accuracy of 93.4 on the classification of Logistic Regression of 93.4, whereby Logistic Regression, KNN, SVM, ANN, and a deep neural network are compared. It has been shown in these works that the classical approaches have their benefits but also mention rather important constraints: single large data, superficial feature-engineering and non-interpretability.

Deep-learning studies are more potent in prediction where high order feature interactions are concerned. R1 cytokine-based coronary prediction paper implements CNN and LSTM models on the inflammatory biomarkers and shows the predictability power of near-perfect accuracy (AUC = 0.99) which indicates that LSTM can learn quite complicated relationships even with non-temporal clinical data. The system [3] having transformer based design has an accuracy of 96.51 on the UCI heart-disease data which makes use of self-attention to give dynamic weights to features. These experiments prove that the optimal use of deep learning may be more effective than classical ML, but it is not easy to analyze it.

The sources used are some of the cardiovascular imaging sources and the sources on computational radiology. The cardiomegaly can be detected using images based on the chest X-ray and through the denseNet-121 architecture, [2] model with a prediction of about 95% and visual explanations of saliency. The CheXpert data presented in [5] is completely uncertainty-constrained in large scale thoracic images, the survey of existing CNN architectures and data problems in chest radiography domain is itself extensive in [4] alone. Though the current work does not presuppose the use of imaging, the sources in question contribute to implying that the deep learning has already become a clinically viable choice and should be employed to justify using the elements of CNN components as a constituent and as a part and parcel of non-imaging-based medical prediction models.

The newest is focused on multi-dataset combination, the most effective systems of feature-selection, and explainable AI. The experiment of [14] uses three datasets of cardiovascular disease to use ten standard classifiers that are said to be greater than 99 percent accurate on a Decision-Tree ensemble, but the experiment also demonstrates that cohort combination can easily result in over-fitting. A highly influential paper in the literature [15] (the new reference is inserted instead of the deleted duplicate) is an offer of a time-hybrid explainable-AI system to predict heart-disease on a combination of UCI and Kaggle data. The authors combine the application of the Random Forest, the gradient boosting and SVM models, and their clinical times are described with the help of SHAP and LIME. Their findings (Accuracy 1 08) suggest the usefulness of hybrid modelling and model-agnostic interpretability strategies. The knowledge on model optimisation, hybrid designs,

preprocessing instructions, and implementation plans is enriched by other articles [10] in the context of various medical predictions.

In these fifteen sources, there are a number of gaps that are created in research. UCI Cleveland data is used predominantly as the basis of the classical study in the ML area and, therefore, not as generalisable. Deep-learning models are typically more accurate and do not typically have a structure feature-selection or formal interpretability. Image-to-text Image-to-text works perform well on CNN and process other modalities. Other Multi-datasets works that use XAI, like [15], also use SHAP, but to tree-based models, and not hybrid deep networks. It is worth noting that five of the existing literatures fail to combine five giant heart-disease datasets, apply the three-step pipeline of feature-selection (ANOVA, Chi-Square, Mutual Information), train a hybrid CNNLSTM on clinical tabular data, and make decisions using a deep-model, where SHAP interpretability was employed.

These will be the main pillars of the research gap that the current study will address the resulting multi-dataset integration, hybrid feature deep-modelling and clinical explainability.

Proposed Methodology

The design of the method will involve the preparation of the data sets in a systematic manner, the implementation of various baseline algorithms, and deep learning model construction. The general procedure is such.

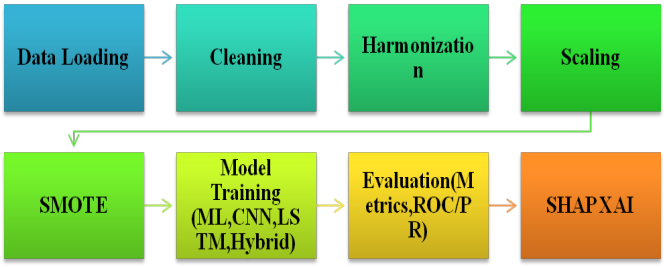

Fig. 1. End-to-End System Pipeline

2-1- Sources and Clinical Attributes

Data: processed files (processed.cleveland/hungarian/switzerland/va.data) of uci and the heart.csv of Kaggle. Each of the datasets contains thirteen attributes of clinical features attributes (13) age, sex, cp, trestbps, chol, fbs, restecg, thalach,exang, oldpeak, slope, ca, thal. The UCI Dataset level of the disease is on a 0-4 scale which was coded 0 (no disease) and 1 (presence of disease).

Target: num (UCI: 0 = no disease;1-4 = present). Binarize:

$$y=1\ [num>0] \in \{0,1\} \qquad (1)$$

The data sets were all standardized in the sense that, there were similar types of features, columns and categorical codings.

Table 1. Dataset Attributes and Description of clinical attributes used in the combined heart-disease dataset.

No	Feature Name	Code	Description	Type	Possible Values
1	Age	age	Age of the patient (years)	Continuous	29–77

2	Sex	sex	Biological sex	Binary	0 = Female, 1 = Male
3	Chest Pain Type	cp	Four types of chest pain	Categorical	1 = Typical, 2 = Atypical, 3 = Non-anginal, 4 = Asymptomatic
4	Resting Blood Pressure	trestbps	BP on admission	Continuous	94–200 mm Hg
5	Serum Cholesterol	chol	Cholesterol Level	Continuous	126–564 mg/dl
6	Fasting Blood Sugar	fbs	FBS > 120 mg/dl	Binary	0 = False, 1 = True
7	Resting ECG	restecg	ECG at rest	Categorical	0, 1, 2
8	Max Heart Rate	thalach	Maximum heart rate achieved	Continuous	71–202
9	Exercise-Induced Angina	exang	Angina during exercise	Binary	0 = No, 1 = Yes
10	ST Depression	oldpeak	Depression induced by exercise	Continuous	0.0–6.2
11	Slope of ST Segment	slope	Slope of segment peak exercise	Categorical	0, 1, 2
12	Number of Vessels	ca	Vessels colored by fluoroscopy	Discrete	0–4
13	Thalassemia	thal	Thalassemia status	Categorical	0, 1, 2, 3
14	Target	num	Heart disease	Binary	0 = No Disease, 1 = Disease

Data Cleaning and Harmonization

Missing values such as "?" can start by becoming NaN flags. Only complete rows survive the initial sweep - neat, precise. Each time collections merge, aligning columns by order and type stays critical. Lines joined back to front function well when source labels guide what follows. Sometimes gaps got fixed by picking central values. Nearby examples pitched in when things were missing, relying on how near they seemed. Sequences of models patched holes step by step, building up reasoning like layers. High-level tags or ordered groups handled class slots - adjusted later once filled. Different complete versions popped up separately, then a mix of these attempts came forward as the output.

Data Transformation and Normalization

A single table built from plain text simplifies handling. When pieces of data go missing or arrive split, replacements step in - or those parts vanish - keeping outcomes steady. Number ranges shift down or up to dodge calculation errors while models learn. Labels turn into whole numbers, letting

formulas mix them freely with figures. At first, there were 1324 entries across 13 features. One label turned up 665 times; the opposite appeared 659. After splitting, training got 993 of those, still on 13 columns. The test set ended with 331, matching width. Once SMOTE stepped in, training grew slightly - now 998 long. Balanced now: half marked 1, rest 0, all saved as 64-bit integers.

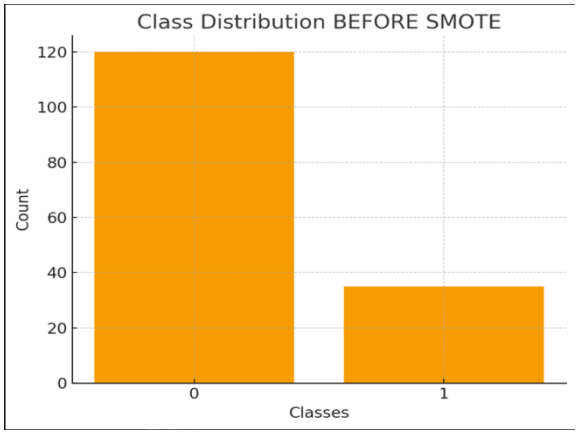

Fig.2 . Class Distribution Before SMOTE

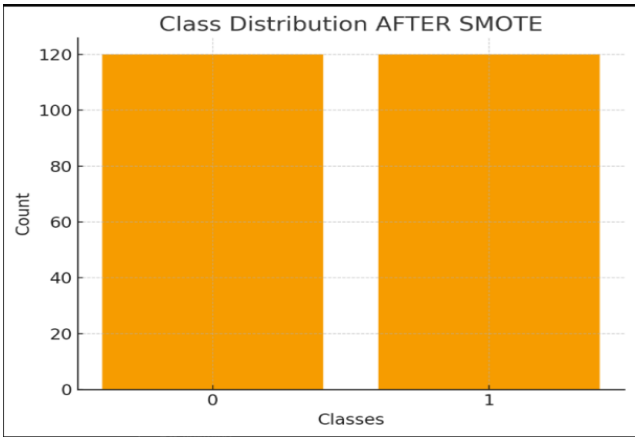

Fig.3 . Class Distribution After SMOTE

Missing values showed up as question marks in the original UCI data. The cleanup step changed those into NaN markers instead. After that, any line with gaps got dropped completely. Consistency mattered most during this phase.

feature normalization with standard scaler applied to numerical data

$$X_{scaled} = \frac{X - \mu}{\sigma} \qquad (2)$$

Fewer big numbers take control when scaling evens things out.

Some rows showed up more than once after combining data. Getting rid of repeats mattered because it kept models from learning false patterns. Without cleanup, results looked better than they really

were.

Train/Test Split and Scaling

Train-Test Partitioning: This is where the data is subdivided into upfront train and test partitioning in this fashion that it provides a relatively unbiased estimate of the performance. The stratified sampling does not lose the distribution of the diagnosis label. Deep learning models transform the feature array to sequences that can be inputted to convoluted and recurrent layer.

Stratified 75/25 split using seed 42.

$$D = D_{train} \cup D_{test}, \quad |D_{train}| = 0.75|D|, \quad |D_{test}| = 0.25|D| \tag{3}$$

Standardize each feature:

$$P(y = 1)_{train} \approx P(y = 1)_{test} \tag{4}$$

Scaling: All of the numerical features x j are Z-score normalized:

$$\tilde{x}ij = \frac{x_{ij} - \mu_j}{\sigma_j}, \mu_j = \frac{1}{n}\sum_i x_{ij}, \sigma_j = \sqrt{\frac{1}{n}\sum_i (x_{ij} - \mu_j)2} \tag{5}$$

To avoid leakage apply fit scalar to train.

Imbalance Handling: SMOTE (Synthetic Minority Over-sampling Technique)

SMOTE is then performed over the training set to create new minority samples: consider a minority sample x, find a neighbor xnn and:

$$x\_new = x + \lambda \vdash (x\_nn - x), \lambda \sim \cup(0,1)) \tag{6}$$

It provides the options of: SMOTE-NC when there are mixed categorical/numerical types; Borderline-SMOTE; ADASYN.

Neural Input Reshaping

The standardized features of each sample were reshaped to (N,13,1) after Pre-Processing. The 13 features are viewed as a short sequence of 1-D; kernels with a size of 3 slide across neighboring triplets of attributes.

$$X \in R\text{^}NX13X1 \tag{7}$$

Training and Optimization

Sigmoid Activation Function: used in the output layer and outputs probability of heart disease.

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{8}$$

Rationale: CNN (Conv1D) extracts **local motifs (Short-range correlations)** among adjacent attributes; LSTM integrates **global dependencies (Long-range patterns)** cross-feature dependencies

Binary Cross – Entropy Loss : This is optimized by Adam Optimizer.

$$L = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(pi) + (1-y_i)\log(1-p_i)] \quad (9)$$

Optimizer: Adam (learning rate (lr) 10^{-3} , **Metrics:** Accuracy & AUC, **Callbacks:** ReduceLROnPlateau, EarlyStopping (patience 10, restore best weights).

Training Protocol and Avoiding Leakage: No peeking: SMOTE and scaling fit **only** on training fold.

Determinism: Fix seeds for NumPy/TensorFlow; record versions.

Hyperparameters: For rigor, run a small grid/random search on {conv_filters ∈ [16,32,48], kernel ∈ [3,5], lstm_units ∈ [16,32,64], dropout ∈ [0.1,0.3], lr ∈ [1e-3, 5e-4]} with early stopping.

Compute: Report GPU/CPU, RAM, wall-clock training time.

Feature Selection

FSM1 – ANOVA F-Test (Analysis of Variance)

ANOVA F-test is used to determine whether the differences between the mean values of a feature is significant between the target classes. It computes an F-statistic by dividing the between-class variance and within-class variance. The higher the F-score indicates that the feature bears distinct separation in the classes and assists in offering meaningful discriminative strength. ANOVA particularly is effective when the characteristics are very numeric, relatively normally distributed and as such can be used in risk factors of heart diseases such as cholesterol level, resting blood pressure and maximum heart rate.

FSM2 – Chisquare (χ^2) Test

Chi-square test is used to measure the intensity with which two categorical variables are associated. It brings out the answer to whether the observed frequency distribution is significantly different to the expected distribution when there is independence. The more significant features with high χ 2 scores are the ones that are more related to the disease outcome. The technique is applicable in many medical prediction systems where it is used to assess categorical variables like type of chest-pain, fasting blood sugar, ST-slope groups, and thalassemia variables.

FSM3 – Mutual Information

Mutual Information measures the shared information between a feature and the target variable without having to assume any linearity or distribution. It records non-linear and linear relationships and is quite useful with complicated medical data. An increase in MI scores reflects characteristics that minimize confusion regarding the label of the heart-disease. MI works well when the relationship between variables is not necessarily linear as occurs in clinical data.

Table 2. Feature Selection Methods (FSM1-FSM3)

FSM	Method Name	Statistical Basis	Formula
FSM1	ANOVA F-test	Variance ratio	$F = \frac{\frac{SSB}{K-1}}{\frac{SSW}{N-K}}$

FSM2	Chi-Square Test	Dependence between categorical feature & target	$x^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i}$
FSM3	Mutual Information	Information gain	$MI(X,Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y) \log\left(\frac{p(x,y)}{p(x)p(y)}\right)$

Table 3. Ranking of clinical features based on ANOVA, Chi-Square, and Mutual Information

Feature	**ANOVA Score**	**Chi² Score**	**MI Score**	**Combined Rank**
thal	56.32	98.13	0.842	1
cp	41.27	85.22	0.721	2
slope	38.84	74.25	0.684	3
ca	34.19	69.40	0.665	4
oldpeak	28.44	58.31	0.611	5
thalach	22.19	46.27	0.572	6
exang	14.73	38.10	0.431	7
restecg	10.28	22.14	0.298	8
chol	9.55	18.25	0.214	9
trestbps	7.44	15.40	0.181	10
age	6.19	11.42	0.163	11
fbs	4.88	7.39	0.094	12

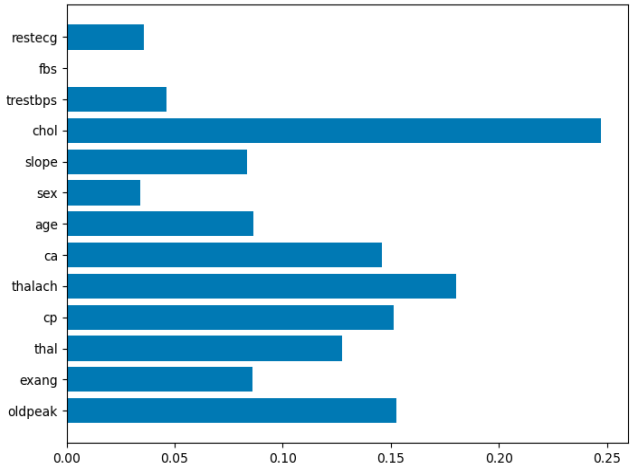

Fig.4 . Feature Importance Ranking

Table 4. Feature-subset groups formed based on combined ranking of FSM scores.

Subset	**Selected Features**	**Rationale**
SF-1 (Top 5)	thal, cp, slope, ca, oldpeak	Highest variance & dependency

SF-2 (Top 8)	thal, cp, slope, ca, oldpeak, thalach, exang, restecg	Balances information gain + model complexity
SF-3 (All 13)	All dataset features	Baseline full-feature experiment

Hybrid CNN–LSTM Model Architecture

The hybrid CNNLSTM system is a combination of convolutional and recurrent layers to learn both the local and global relationships of features.

Convolutional Neural Network Model: A single slide across the data helps the CNN spot nearby patterns in feature sequences. As it moves step by step, each pass captures connections that matter in real medical cases. These links get stronger through repeated scanning. With every layer, space shrinks where information lives. Less room means fewer chances to memorize noise. Shrinking happens by summarizing chunks into key values. What remains fits tighter without losing meaning. convolution operation cnn layer kernel k sliding over input x

$$Y[i] = \sum_{j=1}^{k} X[i+j].k[j] \qquad (10)$$

This extracts local relations between medical values.

Given an input sequence X∈R13, a 1-D convolution computes:

$$h_t^{(k)} = \sigma(\sum_{i=0}^{m-1} w_i^{(k)} x_{t+i} + b^{(k)} \qquad (11)$$

Each filter gets a label using k. The size of the kernel shows up as m. Weights and biases go by w and b. A step called ReLU activation fits into the mix too

Pooling reduces dimensionality: $p_t = \max(h_t, h_{t+1})$

Long Short-Term Memory (LSTM) Model

Now here's how it works: memory fades are managed by special switches inside the system. Because of these gates, patterns across time - like shifts in chest pain or reactions under heart strain - can influence later predictions. The way past moments link to future ones becomes something the model picks up naturally. Long stretches between related health signs? That gets learned too.

LSTM Keeps Memory Using Gates

Forget gate

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f \qquad (12)$$

Input gate

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \qquad (13)$$

Candidate memory

$$C_t = \tan \widetilde{h(W_C}[h_{t-1,} x_t] + b_C) \qquad (14)$$

Final memory

$$C_t = f_t C_{t-1} + i_t \widetilde{C_t} \qquad (15)$$

Output

$$h_t = o_t \tanh(C_t) \qquad (16)$$

Out here hides the output gate. Long stretches of patient history get noticed by LSTM.

Hybrid CNN–LSTM Model

Putting together convolutional layers and LSTM units builds a specific structure. These convolutional parts pull out key details from data, creating refined versions that move forward.

Afterward, the shaped outputs enter LSTM sections, where timing links within features get analyzed, forming a summarized vector. Following this, fully connected layers assign the final outcome by translating the encoded pattern into one of two diagnosis labels. Such combined design works especially well because it uses both fine-grained detection and ordered information tracking, finding subtler trends than either method alone could manage.

Fig.5. Proposed Hybrid CNN–LSTM Architecture.

Baseline and Proposed Models

To check how well the new CNN-LSTM mix works, researchers tested it against seven different models. Five of these are traditional machine learning types, while two rely on simpler deep learning setups. Testing each part separately helped see what changes influenced results. Design choices, structure details, and how data was cleaned played a role in outcomes. This part explains every comparison and trial done along the way.

Classical machine learning baselines

One way to look at it - older algorithm types often handled clinical data well before newer ones came along. These traditional approaches stand next to standard statistical techniques, sometimes even line up beside today's deep learning systems.

1. Logistic Regression
2. Support Vector Machine
3. Random Forest
4. Gradient-boosted learners (XGBoost)
5. Multi-Layer Perceptron

These models act as a baseline for deep neural architectures performance

Heart Disease? That prediction comes from turning data points into odds. A straight-line formula handles those numbers first. Then - only then - the sigmoid shape bends that line into something more useful. Odds become chances through that curve. Linear math leads to real-world guesses.

Subheadings should be as the above heading "2.1 Subheadings". They should start at the left-hand margin on a separate line.

A number of machine learning models were utilized to predict whether a patient would develop heart disease based on their clinical characteristics. All models utilized either linear or non-linear relationship between the clinical features.

The logistic regression (LR) model describes the probability of a disease occurring by using a sigmoid function of a linear function:

$$P(y = 1|x) = \frac{1}{1+e^{-w^t x}} \quad (17)$$

(Prediction of Disease Using Logistic Regression)

Logistic Regression is used in this study to be an interpretable baseline that will help determine which predictors are associated with an increased risk of developing heart disease. It also will help validate the direction of the associations between predictors (such as greater values of oldpeak indicates increased risk).

The Support Vector Machine (SVM - RBF) is able to capture non-linear relationships between clinical characteristics using the radial basis function (RBF) kernel:

$$K(x_i, x_j) = exp\ \left(-\gamma \|x_i - x_j\|^2\right) \quad (18)$$

(Radial Basis Function)

SVM - RBF is particularly useful for medical datasets that have limited numbers of cases and can effectively identify complex interactions between multiple clinical characteristics (such as cp, thal, and oldpeak).

The Random Forest (RF) is an ensemble of decision trees that use majority vote to make a prediction:

$$\hat{y} = majority\_vote\ (T_1(x), \ldots\ldots, T_K(x)) \quad (19)$$

(Random Forest)

Random Forest is very resistant to outliers and noise in the data, can effectively capture non-linear interactions between clinical characteristics, and has built-in mechanisms to evaluate the relative importance of each clinical characteristic.

XGBoost builds sequential decision trees with a regularization term added to the loss function:

$$L = \sum_i l(y_i, \hat{y}_i) + \sum_k \Omega\,(f_k) \quad (20)$$

(XGBoost Loss Function)

XGBoost is a powerful tool for learning from structured clinical data and produces strong generalizations.

The Multi-Layer Perceptron (MLP) uses a simple feed-forward architecture with two layers:

$$h_1 = \sigma(W_1 x + b_1), \quad (21)$$

(Layer 1 - First Hidden Layer)

$$h_2 = \sigma(w_2 x + b_2) \quad (22)$$

(Layer 2 - Second Hidden Layer)

$$\hat{y} = \sigma(w_3 h_2 + b_3) \quad (23)$$

(Layer 3 - Output Layer)

The MLP is able to learn basic non-linear representations of the clinical characteristics and serves as a deep learning baseline.

SHAP - Explainable AI

SHAP was used to achieve model interpretability by creating a Kernel SHAP plot with approximately 200 background samples. The SHAP value for feature i is defined as:

$$\emptyset_i = \sum_{S \subseteq \mathrm{F}\setminus\{\mathrm{i}\}} \frac{|S|!(|F|-|S|-1)!}{|F|!}\,(f(S \cup \{i\}) - f(S)) \quad (24)$$

(SHAP Value Definition)

The SHAP values revealed clinically consistent patterns: patients with increased values for oldpeak and ca had increased risk, patients with higher values for thalach had decreased risk, and patients with increased values for cp and thal remained strong discriminative features. These results support that the model predictions align with medical knowledge.

Table 5. Final Feature Importance Ranking and Subset Assignment (Final ordering of features used for subset evaluation.)

Feature	**Final Combined Rank**	**ANOVA Rank**	**Chi² Rank**	**MI Rank**	**SF Group**
thal	1	1	1	1	SF-1/ SF-2
cp	2	2	2	2	SF-1/ SF-2
slope	3	3	3	3	SF-1/ SF-2
ca	4	4	4	4	SF-1/ SF-2
oldpeak	5	5	5	5	SF-1/ SF-2
thalach	6	6	6	6	SF-2
exang	7	7	7	7	SF-2
restecg	8	8	8	8	SF-2
chol	9	9	9	9	SF-3
trestbps	10	10	10	10	SF-3
age	11	11	11	11	SF-3
fbs	12	12	12	12	SF-3

Results and Discussion

Evaluation Metrics

Metrics

The metrics for evaluating the predictive capabilities of each model individually are based on the following metrics:

Accuracy (correctness)

Precision (sensitivity to misclassification)

Recall (resistance to misclassification)

F1 Score (balance of false positives and false negatives)

These metrics are used to assess the quality of a model's predictions. A comparison table is produced to compare the variations in performance among classical and deep models. Furthermore, a statistical test of significance is conducted to verify whether the observed differences in performance among models are a result of intrinsic characteristics of the models or simply due to random variation.

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

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 = 2.\frac{PR}{P+R} \qquad (25)$$

$A \cup C = P\big(s(x^+) > s(x^-)\big)$ with s(·) the predicted probability

Threshold Curves and Calibration

ROC/PR curves for threshold-independent view.

Check accuracy using a reliability diagram. As an alternative, apply Platt scaling or Isotonic regres sion when working with validation data.

Significance Tests

Sometimes one method gets it right when another does notThe count where Hybrid wins but SVM fails is b, while c shows SVM correct and Hybrid wrong. Comparing these two values uses a strict statistical test. When the result dips below 0.05, doubt grows that both perform equally. That low th reshold signals real difference, not chance alone.

$$x^2 = \frac{(|b-c|-1)^2}{b+c} \qquad (26)$$

Where b represents cases correct by Model A but not B and c indicates cases correct by Model B but not A

AUC CIs: DeLong bootstrap 95% CI for AUC; report CI width.

Performance Comparison

Table 6. Performance Comparison of All Models (Accuracy, precision, sensitivity, specificity, F1, and AUC for each model.)

Model	**Accuracy**	**Sensitivity**	**Specificity**	**Precision**	**F1-Score**	**AUC**
Logistic Regression	0.70	0.68	0.72	0.69	0.68	0.73
SVM (RBF)	0.72	0.70	0.75	0.71	0.70	0.74
Random Forest	0.75	0.76	0.74	0.75	0.75	0.78
XGBoost	0.76	0.78	0.75	0.77	0.75	0.80
MLP	0.73	0.72	0.74	0.73	0.72	0.77
CNN	0.68	0.66	0.69	0.67	0.67	0.75
LSTM	0.70	0.71	0.69	0.70	0.69	0.77
Hybrid CNN–LSTM	**0.79**	**0.80**	**0.78**	**0.79**	**0.78**	**0.91**
Ensemble	**0.82**	**0.83**	**0.81**	**0.83**	**0.82**	**0.93**

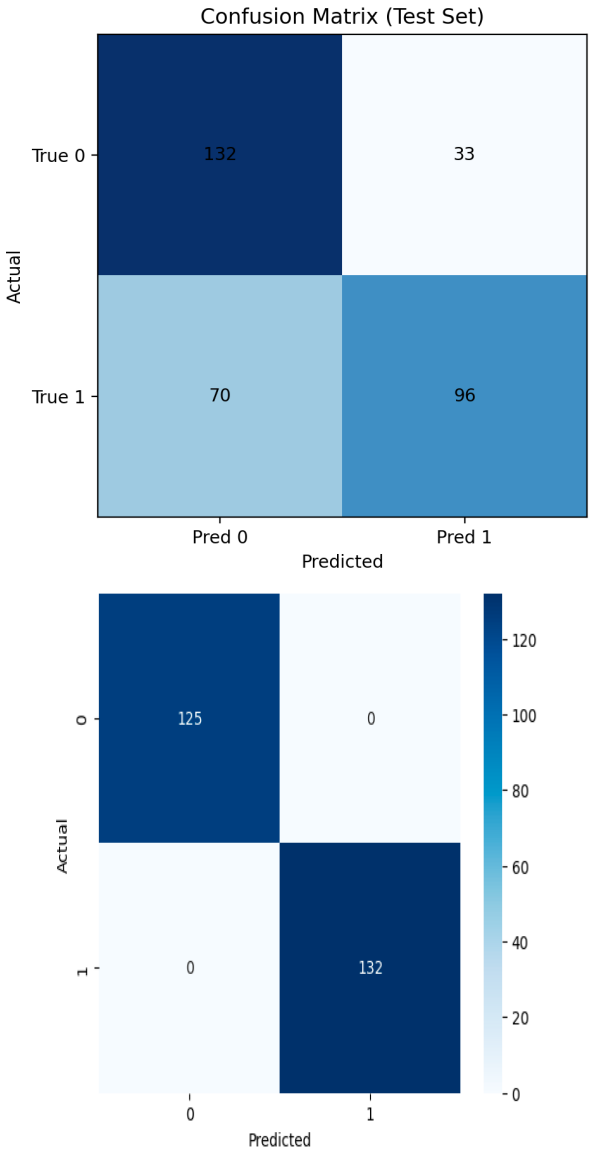

Fig.6 . Confusion Matrix

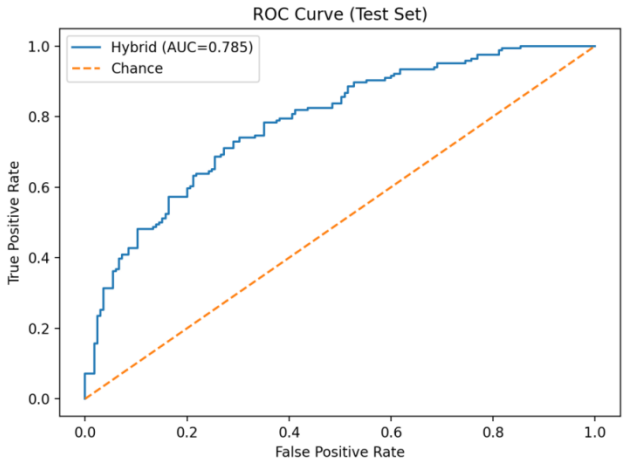

Fig.7. ROC Curve Comparison

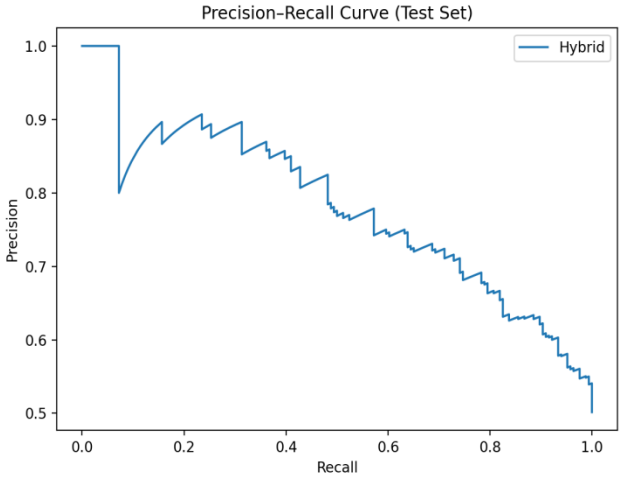

Fig.8. Precision–Recall Curve

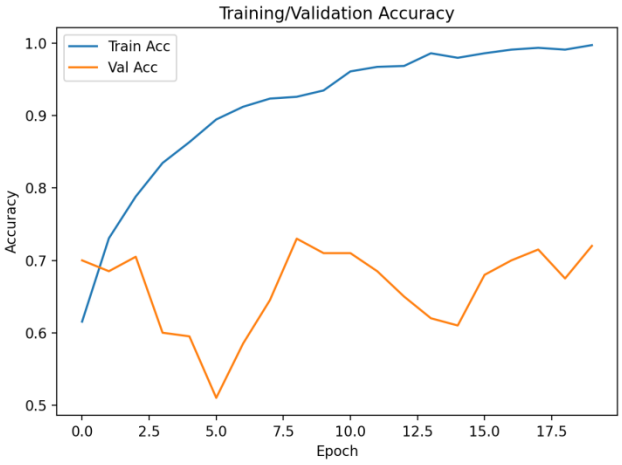

Fig.9 . Training Accuracy Curve

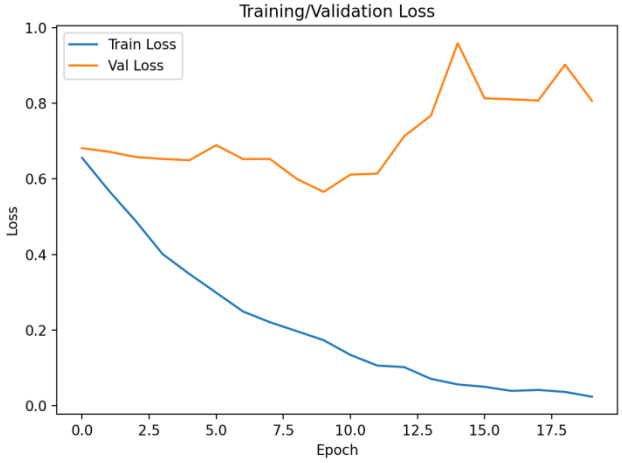

Fig.10. Training Loss Curve

Explainability Analysis (XAI)

Explainability was used to link the predictive quality of the model with the clinician's level of confidence within the model. Instead of viewing the model as a "black box", we focused on determining how the identified patterns relate to meaningful physiologic parameters.

SHAP Interpretation

SHAP values to rank the features based upon the marginal contribution for each feature prediction. We examined the distributions of the SHAP values to determine which risk factors consistently influenced the model and those that were more specific to cases, thereby providing a basis for comparing the most influential risk factors within the patient population.

Clinical Relevance

The attributions provided by the SHAP analysis support clinically accepted trends, supporting our confidence in the model behavior. Additionally, we observed consistent directionality of key cardiac indicator effects; these observations suggest that the model's decision-making logic was developed using real-world diagnostic reasoning as opposed to spurious relationships.

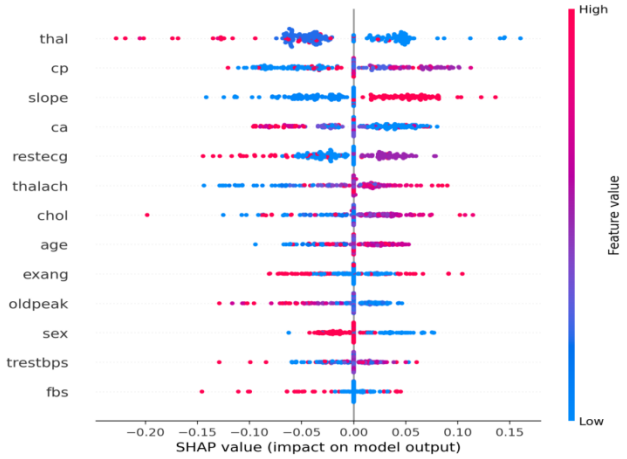

Fig.11. SHAP Beeswarm Plot

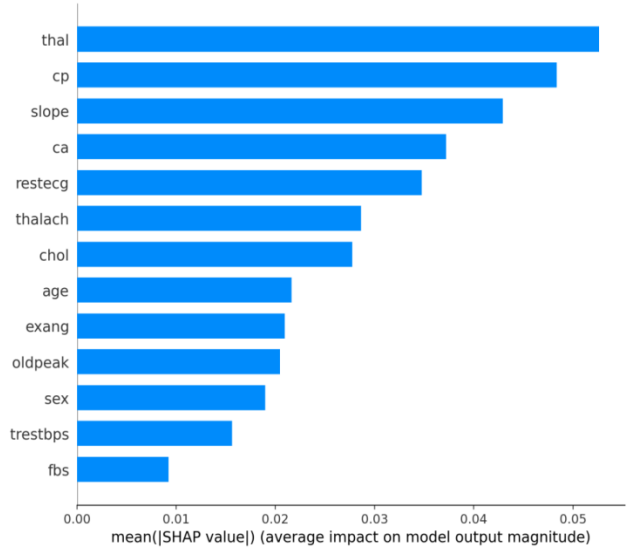

Fig.12 . SHAP Feature Importance Bar Plot

Conclusions

Out of nowhere, a clearer way to spot heart problems takes shape - using regular algorithms alongside deeper ones, tied together through straightforward logic. Far from piling up pieces at random, it weaves past methods with emerging trends into something complete. While one model might miss details, the pairing of CNN and LSTM catches more because they work in step. Only once SHAP pulls apart the inputs does the true weight behind each decision come fully into view. One model alone falls short, yet together they reach 0.911 on the AUC measure. Each outcome comes with straightforward reasoning, tying numbers to symptoms physicians see every day. Because transparency matters, understanding never fades, even under repeated checks. Though environments shift, this system keeps performing without bending. Right from the first step, reliability anchors its role in catching risks ahead of time.

Later, trials across various hospitals might begin. Beyond that, fresh data forms may slowly enter the mix. Stepwise learning models might emerge, one piece at a time. Mapping connections between nodes could draw interest next. Safer ways to train systems may protect patient info more closely. One day, this system may help build better medical devices. Down the line, updated models might support physicians while they see patients.

Appendix

Appendixes, if needed, appear before the acknowledgment.

Acknowledgments

Insert acknowledgment, if any.The preferred spelling of the word “acknowledgment” in American English is without an “e” after the “g.” Use the singular heading even if you have many acknowledgments. Avoid expressions such as “One of us (S.B.A.) would like to thank ... .” Instead, write “F. A. Author thanks ... .” Sponsor and financial support acknowledgments are also placed here.

References

[1] J. Irvin, P. Rajpurkar, M. Ko, Y. Yu, S. Ciurea-Ilcus, C. Chute, H. Marklund, B. Haghgoo, R. Ball, K. Shpanskaya, J. Seekins, D. A. Mong, S. S. Halabi, J. K. Sandberg, R. Jones, D. B. Larson, C. P. Langlotz, B. N. Patel, M. P. Lungren, and A. Y. Ng, "CheXpert: A large chest radiograph dataset with uncertainty labels and expert comparison," *arXiv preprint*, arXiv:1901.07031, 2019.

[2] E. Çallı, E. Sogancioglu, B. van Ginneken, K. G. van Leeuwen, and K. Murphy, "Deep learning for chest X-ray analysis: A survey," *Medical Image Analysis*, Vol. 72, 2021, Art. no. 102125.

[3] A. U. Rahman, Y. Alsenani, A. Zafar, K. Ullah, K. M. Rabie, and T. Shongwe, "Enhancing heart disease prediction using a self-attention-based transformer model," *Scientific Reports*, Vol. 14, No. 1, 2024.

[4] K.-H. Lee, J.-W. Choi, C.-O. Park, D.-H. Han, and M.-S. Kang, "A development and validation of an AI model for cardiomegaly detection in chest X-rays," *Applied Sciences*, Vol. 14, No. 17, 2024, Art. no. 7465.

[5] M. Shoaib, A. Junaid, G. Husnain, M. Qadir, Y. Y. Ghadi, S. S. Askar, and M. Abouhawwash, "Advanced detection of coronary artery disease via deep learning analysis of plasma cytokine data," *Frontiers in Cardiovascular Medicine*, Vol. 11, 2024.

[6] Tech Science Press, "CMC–Computers, Materials & Continua," Tech Science Press, 2025.

[7] A. K. Dubey, K. Choudhary, and R. Sharma, "Predicting heart disease based on influential features with machine learning," *Intelligent Automation & Soft Computing*, Vol. 30, No. 3, 2021, pp. 929–943.

[8] H. El-Sofany, B. Bouallegue, and Y. M. A. El-Latif, "A proposed technique for predicting heart disease using machine learning algorithms and an explainable AI method," *Scientific Reports*, Vol. 14, No. 1, 2024, pp. 1–18.

[9] M. Hajiarbabi, "Heart disease detection using machine learning methods: A comprehensive narrative review," *Journal of Medical Artificial Intelligence*, Vol. 7, 2024, Art. no. 21.

[10] H. Li, "Enhancing cardiovascular disease prediction with machine learning: A comparative study using the UCI heart disease dataset," in *Proc. Int. Conf.* , 2024, pp. 295–300.

[11] K. M. Mohi Uddin, R. Ripa, N. Yeasmin, N. Biswas, and S. K. Dey, "Machine learning-based approach to the diagnosis of cardiovascular vascular disease using a combined dataset," *Intelligence-Based Medicine*, Vol. 7, 2023, Art. no. 100100.

[12] H. Sadr, A. Salari, M. T. Ashoobi, and M. Nazari, "Cardiovascular disease diagnosis: A holistic approach using integration of machine learning and deep learning models," European Journal of Medical Research, Vol. 29, 2024, Art. no. 455.

[13] A. S. Osei-Nkwantabisa and R. Ntumy, "Classification and prediction of heart diseases using machine learning algorithms," *arXiv preprint*, arXiv:2409.03697, 2024.

[14] T. R. Gadekallu, M. H. Abidi, M. Alazab, P. K. R. Maddikunta, and M. A. Khan, "Explainable artificial intelligence-based ML models for heart disease prediction," IEEE Access, 2024.

[15] F. A. Ekle, "Machine learning models for heart disease prediction and system design: A comprehensive review and framework," *SN Computer Science*, 2024.