ISSN: 2182-2069 / E-ISSN: 2182-2077

A Privacy-Preserving Cloud-Based System for Cardiovascular Disease Prediction Using Lightweight Machine Learning Models

D.Cenitta[1], Seema Patil[2], K. Asha[3], M. Anitha[4], T. Sowmya[5], Prameetha Pai[6*], and R. Vijaya Arjunan[7*]

[1]Department of Computer Science and Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, Karnataka, India. cenitta.d@manipal.edu, https://orcid.org/0000-0003-3715-6941

[2]Assistant Professor, Department of CSE, B.M.S. College of Engineering, Bengaluru, Karnataka, India. seemapatil.cse@bmsce.ac.in, https://orcid.org/0000-0001-6490-3020

[3]Assistant Professor, Department of AIML, Dayananda Sagar College of Engineering, Bengaluru, Karnataka, India. ashakcse04@gmail.com, https://orcid.org/0009-0000-4458-6806

[4]Assistant Professor, Department of CSE, Dayananda Sagar College of Engineering, Bengaluru, Karnataka, India. anitha-cs@dayanandasagar.edu, https://orcid.org/0000-0003-2297-3180

[5]Assistant Professor, Department of CSE, B.M.S. College of Engineering, Bengaluru, Karnataka, India. sowmyat.cse@bmsce.ac.in, https://orcid.org/0009-0005-3971-7791

[6*]Assistant Professor, Department of CSE, B.M.S. College of Engineering, Bengaluru, Karnataka, India. prameethapai.cse@bmsce.ac.in, https://orcid.org/0000-0003-2526-2931

[7*]Department of Computer Science and Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, Karnataka, India. vijay.arjun@manipal.edu, https://orcid.org/0000-0002-1402-6573

Received: April 17, 2025; Revised: May 30, 2025; Accepted: July 21, 2025; Published: August 30, 2025

Abstract

Since cardiovascular disease (CVD) continues to be a major cause of death worldwide, early and precise prediction tools are essential. The results of this work point to an architecture that runs ML models in the cloud on a user's data, forecasting the risk of CVD in real time, and is mainly useful in resource-constrained environments. It applies privacy-ensuring techniques when handling patients' data as it is transmitted and used, and it relies on secure cloud computing for managing the information on a large scale. When the aim is to use fewer computer resources without losing accuracy, lightweight models are preferred, for instance, ensemble methods and efficient decision trees. How accurate and precise it is, the quality of its F1-score and how efficiently it runs are assessed once benchmark clinical datasets have been used to validate it. Since the suggested

Journal of Internet Services and Information Security (JISIS), volume: 15, number: 3 (August), pp. 129-148.
DOI: *10.58346/JISIS.2025.I3.009*

*Corresponding author: Assistant Professor, Department of CSE, B.M.S. College of Engineering, Bengaluru, Karnataka, India; Department of Computer Science and Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, Karnataka, India.

approach is effective, reliable and secure, it works well for mHealth apps and remote healthcare, the findings indicate.

Keywords: Cardiovascular Disease Prediction, Privacy-Preserving Machine Learning, Cloud Computing, Lightweight ML Models, Mobile Health (mHealth), Secure Data Transmission, Healthcare Analytics, Edge-Cloud Integration.

1 Introduction

In line with the World Health Organization, CVDs cause over 17.9 million deaths every year which equals to 32% of total worldwide deaths and makes CVDs a major global concern (World Health Organization, 2009). Some of these conditions such as heart failure, cerebrovascular illness and coronary artery disease, are heart and blood vessel problems. It is very important to catch serious diseases early and keep an eye on patients afterward to reduce mortality and improve outcomes. Unfortunately, many people in rural or underpowered areas have little access to expert cardiologists or advanced diagnosis technology. The application of machine learning has helped in creating algorithms that scan medical data and easily predict the risk of heart disease (Javed et al., 2023). It is sometimes hard for clinicians to notice patterns and links in patient data without the help of computers. When used with cloud computing infrastructure, intelligent systems can be used in many different parts of the world. Using cloud systems is perfect for running predictive healthcare applications since they offer easy central storage, access and scalability (Wang et al., 2024).

But there are major challenges related to privacy, security and running machine learning systems well, when using the cloud in healthcare. If health data is sent to the cloud without proper safety rules, it can lead to breaking laws such as HIPAA and GDPR (Protection, 2018). At the same time, advanced machine learning methods are accurate, but tend to be too demanding and thus not suitable for applications on mobile devices. There is a greater need for solutions that forecast accurately, rely on simple models and protect people's data from end to end. To meet this requirement, we suggest a cloud service for mHealth and edge computing that keeps patient data safe by using lightweight machine learning models to find early indications of CVD.

Previously, many studies have investigated the use of algorithms for CVD prediction (Sharma & Rajput, 2024). The authors defined in (Singh & Kumar, 2020) used health data from UCI to identify people with heart disease using Random Forests and Support Vector Machines with a high level of accuracy. In a similar fashion, Sundarasen et al. (Wang et al., 2024) made the forecasts of CVDs more robust with ease by using ensemble learning. In most cases, research today gives more importance to making models accurate rather than reducing computational loads and ensuring data privacy. It may not be possible to add these machine learning models directly to hardware used in clinical settings or app environments because of limitations. Many newer methods now employ federated learning which involves training models locally and pooling the results remotely, to deal with privacy problems (Yang et al., 2019). There are those who address data leaks by applying differential privacy or homomorphic encryption (Gentry, 2009). These security methods may work well but are not practical for massive use or in real-time situations because they cost too much to operate (Nawaz & Ahmed, 2022).

For the system to work as planned, using lightweight machine learning is very important. Certain algorithms do well on standard processors, taking up little memory, differently from deep learning models that must be run on GPUs and need large memory (Ke et al., 2017). It matters a lot for IoT and mobile applications, since how quickly an app responds and how long the battery lasts are very important. Because it uses advanced ways to choose the best features and weigh every instance, V-

AdaBoost surpasses ordinary boosting methods by speeding up the learning process and raising its accuracy. The model in this paper is checked against official diagnoses once it has been trained on clinical information. It has been proven that lightweight models are suitable for real-time systems and perform as well as larger models when it comes to prediction.

One of the main challenges for cloud-based healthcare is to stop unwanted access and guarantee transparency in handling data. Disruptions in healthcare systems may bring about identity theft, fraud with health insurance and make people less confident in the system. Due to this fact, our system relies on different security measures (Karimov & Sattorova, 2024). No matter if the data is transmitted or stored, it is all encrypted. People using the system are only allowed if they have secure API tokens and are on the access control list (ACL). Furthermore, audit logs monitor and keep a record of any activity that involves processing or access, assisting with following regulations (Vanderpool, 2019; Shokri & Shmatikov, 2015). These measures make certain that the system is both ethical and technically sound when managing healthcare data (Rao & Saxena, 2025).

Linking edge and cloud environments is also a major part of the method we suggest. A few important steps, for example feature extraction and data validation, are handled by the client machine, while the actual model for prediction is kept in the cloud to make use of more processing power. A mixture of this strategy enables the system to react faster, cut down the number of delays and operate normally even with occasional changes in internet access (Satyanarayanan, 2017). Because of this, whenever new features or models are added, the cloud handles improvements without the need for updates from users. The architecture of today's digital healthcare ensures that it can expand with growth, runs smoothly and is easy for users (Basu & Muthukrishnan, 2024).

Even though it offers certain benefits, the suggested system does have many problems. It is a concern that the model might not apply the same way to different groups of patients and doctors. Most of the data used in machine learning research belongs to specific groups which can limit the usefulness of the models in other situations or for other people. Additionally, human error or incomplete submissions can cause large variations in the quality of input data in real-world applications. The dependence on centralized cloud infrastructure, which could be dangerous in the event of outages or connectivity issues, is another drawback. To enable decentralized training and updating of models directly on user devices, future research can investigate the usage of federated learning (Kairouz et al., 2021). Additionally, enhancing clinician trust and guaranteeing that model predictions are intelligible and actionable will require the integration of explainable AI (XAI) techniques (Lundberg & Lee, 2017).

2 Related Work

Because machine learning (ML) models have the potential to improve diagnostic accuracy and facilitate early intervention, many studies have investigated the prediction of cardiovascular diseases (CVDs) using ML approaches. Clinical datasets like the UCI Heart Disease dataset have historically been subjected to models like Decision Trees, Naïve Bayes, Support Vector Machines (SVM), and ensemble approaches. In their comparison of Random Forest, SVM, and Logistic Regression models, Dey et al. showed that Random Forest outperformed SVM in detecting heart disease patients (Pillai & Bhatia, 2024). But if not optimized, these models can be computationally costly and could not work well in cloud-deployed or mobile settings.

To achieve high classification accuracy, Sundarasen et al. (Yazdi & Asadi, 2025) suggested an ensemble-based CVD prediction model that combines multiple base learners. However, their methodology did not address data privacy or contexts with limited resources. Shadab et al. (Miah et al.,

2023) highlighted the necessity of striking a balance between computing efficiency and prediction accuracy in a different study, proposing the usage of lightweight classifiers such as Decision Trees and Logistic Regression in low-power devices.

It is now possible to outsource CVD prediction activities to cloud platforms thanks to recent advancements in cloud computing. A cloud-enabled telecardiology system for remote diagnosis using ECG data was presented by Zhou et al. (Sahu et al., 2021). This work demonstrated the scalability of cloud computing in healthcare, but it was susceptible to data breaches since it lacked a tangible implementation of privacy-preserving measures. The use of cloud-based intelligent agents for healthcare applications was also done by Tazehkand et al. (Rani & Baburaj, 2019), although their method did not use lightweight models, which led to a longer inference time and higher energy usage.

In ML-based healthcare systems, data security and privacy have emerged as crucial areas of attention. Federated learning approaches in healthcare were inspired by Shokri and Shmatikov's (Jindal et al., 2021) introduction of a privacy-preserving deep learning framework through distributed model training. Federated learning improves privacy by enabling ML models to be trained across dispersed devices without sending private information. In their review of federated learning in healthcare applications, Yang et al. (Yang et al., 2019) noted several significant obstacles, including non-IID (non-independent and identically distributed) data and communication overhead.

Differential privacy is another well-liked technique to improve privacy. A rigorous mathematical methodology was put forth by Dwork (Dwork, 2006) to guarantee that the output of a query is not substantially impacted by the inclusion or deletion of a single data piece. Still, using differential privacy usually adds some noise to the data which can worsen model results in situations where the dataset is very small, as in medical records. In Gentry's findings (Gentry, 2009), it is possible to process data that is encrypted without first having to decrypt it. Even though it offers strong security, this approach takes too much time to work for real-time prediction in CVD cases.

Lately, more attention has been given to light machine learning models since they perform well and are suitable for edge and mobile environments. The Light GBM algorithm, which is very efficient for real-time applications, was described in (Chen & Guestrin, 2016) by Chen et al. The research showed that the key to high performance and accuracy is not to complicate the model too much or select all the unnecessary features. Bhan et al. (Bhan et al., 2023) were able to classify heart diseases probabilities with high percentages and low energy usage through segmenting cardiovascular images (More & Sen, 2024; Lundberg & Lee, 2017)

New research has continued to investigate this subject. Kumar et al (Prasad et al., 2024) proposed a multi-step deep learning approach and explained that using a simpler model makes predictions faster. In their approach, called 'Decentralized Collaborative Healthcare Prediction', Li et al. (Kumar et al., 2024) introduced a privacy-preserving strategy that enables patient data to be kept private as they learn together. To reduce energy and latency, Ahmed et al. (Wang et al., 2024) presented an edge-cloud approach for identifying heart diseases that worked effectively. Zhang et al. evaluated and compared groups of lightweights which help reduce the memory required in cell phones, in remote health monitoring settings (Jeon et al., 2020). Finally, Das and Roy (Iqbal et al., 2024) reviewed whether lightweight classifiers could be linked with explainable AI to help make clinical decisions more interpretable while not increasing the computation needed.

By combining cloud-based architecture, privacy-preserving methods, and lightweight machine learning models into a single system for CVD prediction, our suggested work expands on these discoveries. In contrast to previous research, our approach maintains quick and precise prediction

performance on a scalable cloud platform while guaranteeing the safe transfer and storage of patient data through encryption and access control. By concentrating on the relationship between privacy, model efficiency, and practical deployability in mobile and cloud healthcare settings, it overcomes the shortcomings of earlier research.

3 Materials and Methods

3.1 Dataset Description

The UCI Heart Disease dataset, a renowned standard for creating and evaluating predictive models for cardiovascular disorders, is used in this study. The dataset has 14 variables and 303 instances, including diagnostic measurements (e.g., resting ECG, maximum heart rate, ST depression, number of main vessels), clinical features (e.g., type of chest pain, resting blood pressure, cholesterol), and demographic features (e.g., age, sex). The binary target variable indicates if cardiac disease is present (1) or not (0). Continuous variables were standardized using Min-Max scaling, and categorical variables like "chest pain type" and "thalassemia" were encoded using one-hot encoding to enhance model performance. Median imputation was used to address missing values. The UCI Heart Disease dataset's summary of features, including descriptions, data formats, and value ranges or categories used for model training and prediction, is shown in Table 1.

Table 1: Summary of Features in the UCI Heart Disease Dataset, Including Descriptions, Data Types, and Value Ranges or Categories Used in Model Training and Prediction

Feature Name	Description	Type	Value Range / Categories
Age	Age of the patient	Numerical	29–77 years
Sex	Gender of the patient	Categorical (binary)	0 = Female, 1 = Male
Chest Pain Type	Type of chest pain	Categorical	0 = Typical Angina, 1 = Atypical, 2 = non-anginal, 3 = Asymptomatic
Resting BP	Resting blood pressure (in mm Hg)	Numerical	94–200
Cholesterol	Serum cholesterol (in mg/dl)	Numerical	126–564
Fasting BS	Fasting blood sugar > 120 mg/dl	Categorical (binary)	0 = False, 1 = True
Resting ECG	Resting electrocardiographic results	Categorical	0 = Normal, 1 = ST-T abnormality, 2 = Left ventricular hypertrophy
Max HR	Maximum heart rate achieved	Numerical	71–202
Exercise Angina	Exercise-induced angina	Categorical (binary)	0 = No, 1 = Yes
Old peak	ST depression induced by exercise	Numerical	0.0–6.2
ST _Slope	Slope of the peak exercise ST segment	Categorical	0 = Upsloping, 1 = Flat, 2 = Down sloping
Num Major Vessels	Number of major vessels (0–3) colored by fluoroscopy	Numerical (discrete)	0–3
Thalassemia	Blood disorder (thalassemia type)	Categorical	1 = Normal, 2 = Fixed defect, 3 = Reversible defect
Target	Diagnosis of heart disease	Categorical (binary)	0 = No disease, 1 = Disease present

3.2 System Architecture

It is explained in Figure 1 that the proposed system has three main parts: the client layer, the cloud processing layer and the security layer. With the use of lightweight deep learning models, this system becomes more scalable, respond in real time and puts robust controls on data privacy.

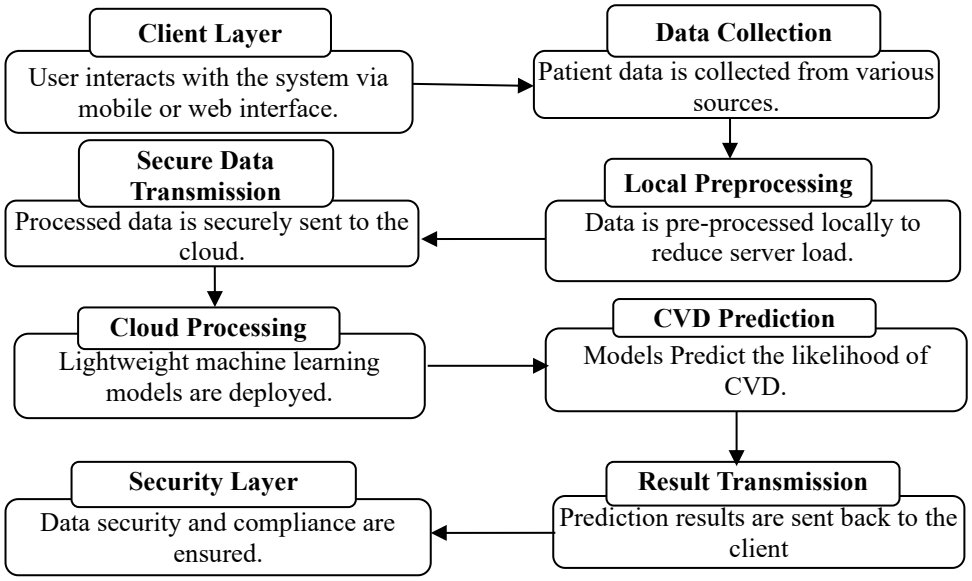

Figure 1: The three-tier architecture of the proposed CVD prediction system, comprising the client interface for data entry and preprocessing, cloud APIs for model inference, and a security layer to ensure privacy and compliance

3.2.1 Client Layer

This layer resides on the user's device, such as a mobile phone or web interface. The client application is responsible for:

- Collecting patient data (e.g., age, cholesterol, blood pressure, ECG results).
- Performing local preprocessing: categorical encoding, normalization, and basic data validation.
- Sending the processed data securely to the cloud layer using HTTPS.
- This reduces the computational burden on the server and improves responsiveness.

3.2.2 Cloud Processing Layer

This layer uses both Flask and Python to move the lightweight ML models to the cloud (on AWS EC2 as RESTful APIs). The model calculates the risk of heart disease for the client after taking their information into account. This layer is meant for fast processing and easy scalability, and it works with several models, for example V-AdaBoost, Decision Tree and Logistic Regression. Because of its efficiency and reliability, the V-AdaBoost method is chosen for structured data. It works with problems that are not easy to classify and forms a powerful classifier using combinations of weaker ones.

Let:

- $D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$ be the training dataset, where $x_i \in \mathbb{R}^n$, and $y_i \in \{0,1\}$.

- w_i be the weight of instance i, initialized as:

$$w_i^{(1)} = \begin{cases} \frac{1}{2n}, & \text{if } y_i = 0 \\ \frac{1}{2k}, & \text{if } y_i = 1 \end{cases} \quad (1)$$

Where n is the number of negative instances and k is the number of positive instances.

For each iteration $t = 1, \ldots, T$:

1. Train weak classifier $h_t(x)$ using weights $w_i^{(t)}$.
2. Compute weighted classification error:

$$\varepsilon_t = \sum_{i=1}^{m} w_i^{(t)} \cdot \mathbb{I}(h_t(x_i) \neq y_i) \quad (2)$$

3. Compute classifier weight:

$$\alpha_t = \frac{1}{2} \ln\left(\frac{1-\varepsilon_t}{\varepsilon_t}\right) \quad (3)$$

4. Update instance weights:

$$w_i^{(t+1)} = \frac{w_i^{(t)} \cdot \exp(-\alpha_t y_i h_t(x_i))}{Z_t} \quad (4)$$

5. Where Z_t is a normalization factor ensuring $\sum w_i^{(t+1)} = 1$.

Final strong classifier:

$$H(x) = \text{sign}(\sum_{t=1}^{T} \alpha_t \, h_t(x)) \quad (5)$$

By emphasizing misclassified samples more and effectively learning from minority trends, this method increases sensitivity to true positive cases, which are crucial for medical diagnosis. Equations (1) through (5) describe the core workflow of the V-AdaBoost algorithm, starting with the initialization of sample weights, iterative error evaluation, weak learner weighting, and weight updating to emphasize misclassified instances, ultimately leading to the final strong classifier.

3.2.3 Security Layer

This layer ensures data security, confidentiality, and regulatory compliance through:

- AES-256 encryption of all data in transit and at rest.
- Token-based user authentication to control access.
- Role-based access controls for API and database management.
- Data anonymization and pseudonymization before model processing.
- Auto-deletion policies and audit trails to comply with GDPR and HIPAA standards.

The system guarantees a reliable and privacy-preserving CVD prediction pipeline that is effective, scalable, and appropriate for implementation in actual mobile health settings by combining these three layers.

3.3 Machine Learning Models

This research uses a collection of lightweight machine learning (ML) models that are appropriate for cloud deployment and real-time inference to enable effective and precise cardiovascular disease (CVD) prediction. These models were chosen due to their competitive accuracy, low latency, and low computational resource consumption, particularly in structured clinical datasets. Four models are used: Versatile Adaptive Boosting (V-AdaBoost), Naïve Bayes, Decision Trees, and Logistic Regression. Below is a description of each model.

3.3.1 Logistic Regression (LR)

A logistic (sigmoid) function is used in the linear classification model known as logistic regression to forecast the likelihood of a binary outcome. It assumes that the log-odds of the target variable and the input features have a linear relationship. It is the perfect baseline classifier for clinical decision-making because it is very interpretable and computationally efficient.

The prediction function is defined as:

$$P(y = 1|x) = \sigma(\mathrm{w}^T\mathrm{x} + b) = \frac{1}{1+e^{-(\mathrm{w}^T\mathrm{x}+b)}} \quad (6)$$

Where:

- $\mathrm{x} \in \mathbb{R}^n$ is the input feature vector,
- $\mathrm{w} \in \mathbb{R}^n$ are the model coefficients,
- b is the bias term,
- $\sigma(z)$ is the sigmoid activation function.

The model is trained using cross-entropy loss and optimized with gradient descent. It performs well with linearly separable data and provides probabilistic outputs that aid interpretability.

3.3.2 Decision Tree (DT)

A non-parametric model called the Decision Tree classifier divides the feature space into sections according to learned decision rules. Each internal node tests a feature, each branch indicates the test's result, and each leaf node denotes a class label in its tree-like structure.

The splitting at each node is based on impurity reduction measures such as Gini Index or Information Gain (Entropy). The Gini index is defined as:

$$Gini(D) = 1 - \sum_{k=1}^{K} p_k^2 \quad (7)$$

Where:

D is the dataset at node,

p_k is the proportion of samples belonging to class k.

Decision Trees can capture non-linear relationships and are interpretable but prone to overfitting. To mitigate this, pruning techniques and depth constraints are applied.

3.3.3 Naïve Bayes (NB)

A probabilistic classifier based on Bayes' Theorem, Naïve Bayes makes the "naïve" assumption that characteristics are conditionally independent. Despite its simplicity, it is incredibly quick throughout the training and prediction stages and frequently performs well with high-dimensional data.

Bayes' Theorem is given by:

$$P(y|x_1, x_2, \dots, x_n) = \frac{P(y)\prod_{i=1}^{n} P(x_i|y)}{P(x_1,x_2,\dots,x_n)} \quad (8)$$

In practice, the denominator is constant for all classes, so classification is based on the numerator only:

$$\hat{y} = \underset{y}{\operatorname{argmax}} P(y)\prod_{i=1}^{n} P\,(x_i|y) \quad (9)$$

Continuous features are modeled using Gaussian distributions, while categorical features use frequency-based estimates. Naïve Bayes is particularly suitable for mobile environments due to its minimal memory footprint. Equations (6) to (9) represent the prediction functions for Logistic Regression, Decision Tree (via Gini Index), and Naïve Bayes, capturing how probabilities and splits are computed for binary classification.

3.3.4 Versatile Adaptive Boosting (V-AdaBoost)

Adaptive Boosting (AdaBoost) is an ensemble learning technique that iteratively focuses on misclassified samples to combine several weak learners (usually decision stumps) into a powerful classifier. To increase convergence speed and accuracy while lowering computing costs, the Versatile AdaBoost (V-AdaBoost) variation employed in this study integrates improved feature weighting and selective sampling techniques.

The V-AdaBoost algorithm operates as follows:

- Initialize uniform weights for each training instance.
- For each round t, train a weak classifier $h_t(x)$ on the weighted data.
- Compute error ε_t and weight α_t of the classifier.
- Update instance weights to emphasize misclassified examples.
- Aggregate weak classifiers into a final strong hypothesis.

Final classifier:

$$H(x) = \text{sign}(\textstyle\sum_{t=1}^{T} \alpha_t\, h_t(x)) \quad (10)$$

This model performs well with imbalanced data, which is a prevalent feature of medical datasets, and is resilient to noise. To cut down on training time and prevent overfitting, V-AdaBoost further provides adaptive feature selection and intelligent stopping criteria. When combined, these four models offer a fair compromise between accuracy, efficiency, and interpretability. While Decision Tree and V-AdaBoost provide better accuracy and resilience to data anomalies, Logistic Regression and Naïve Bayes provide quick and clear baselines. Lightweight and modular classifiers allow for mobile deployment and cloud scalability in real-time health monitoring systems without compromising diagnostic quality.

3.4 Feature Selection and Optimization

Reducing overfitting, enhancing model generalization, cutting down on computation time, and boosting interpretability all depend on effective feature selection, particularly in real-time and resource-constrained cloud or mobile scenarios. The two-stage method used in this study is Grid Search for hyperparameter optimization and Recursive Feature Elimination (RFE) for feature selection.

3.4.1 Recursive Feature Elimination (RFE)

By recursively training a model and eliminating the least significant features depending on the model's performance, Recursive Feature Elimination (RFE) is a wrapper-based feature selection technique that finds the most pertinent features. Until the ideal subset of traits is found, this backward selection procedure is repeated. The following are the main steps of RFE:Train the model (e.g., Logistic Regression or Decision Tree) on the entire set of features.

1. Compute feature importances (weights or Gini scores).
2. Remove the least important feature.
3. Repeat steps 1–3 until a predefined number of features remain.
4. Let $F = \{f_1, f_2, \dots, f_n\}$ be the original feature set. RFE selects a subset $F' \subset F$ such that:

$$F' = \underset{F_k \subset F}{\operatorname{argmax}} \text{Performance}(Model|F_k) \qquad (11)$$

To adjust the feature selection to the model's learning behavior, this procedure is performed for every machine learning model. The ideal subset in our tests often contained 8–10 characteristics that had a major impact on the prediction accuracy (e.g., age, kind of chest discomfort, resting blood pressure, cholesterol, and maximal heart rate). Equations (10) and (11) emphasize ensemble learning and Recursive Feature Elimination (RFE), respectively, for combining classifiers and selecting optimal feature subsets.

3.4.2 Hyperparameter Tuning via Grid Search

Following feature selection, each ML model's optimal hyperparameter configuration is determined via Grid Search. Using 5-fold cross-validation, Grid Search thoroughly assesses every potential combination of the parameters that are supplied and chooses the one that performs the best on a validation set. Let:

- $\Theta = \{\theta_1, \theta_2, \dots, \theta_m\}$ be the hyperparameter space.
- $M(\theta_i)$ be the model trained with hyperparameter θ_i.

Then the optimal parameter set is:

$$\theta^* = \underset{\theta \in \Theta}{\operatorname{argmax}} \text{CrossValScore}(M(\theta)) \qquad (12)$$

The following hyperparameters were optimized:

- Logistic Regression: Regularization type (L1, L2) and penalty coefficient C.
- Decision Tree: Maximum tree depth, minimum samples per leaf.
- Naïve Bayes: Laplace smoothing parameter α.
- V-AdaBoost: Number of estimators, learning rate, and base estimator depth.

Five-fold cross-validation was used to assess each model's resilience to bias and overfitting brought on by data partitioning. The final models for deployment in the cloud-based prediction system were then trained using the chosen feature set and ideal hyperparameters.

3.4.3 Computational Efficiency

Depending on the model, training time was decreased by roughly 30–50% by optimizing hyperparameters and lowering feature dimensionality. 20–35% improvement in inference time is crucial for real-time mobile apps. Additionally, RFE improved generalization by assisting in the removal of noisy or duplicated features (such as old peak or Fasting BS when irrelevant for models). Equation (12) captures hyperparameter tuning using grid search to identify the most effective model configuration.

3.5 Privacy-Preserving Mechanisms

It is essential for a cloud-based prediction system to guarantee privacy, mainly because healthcare data is very sensitive. Different security levels and methods are in place in the suggested system to ensure that data is kept private, safe and accessible only to allowed people as required by international data protection laws, including the GDPR and HIPAA. All the main parts of handling data—transmission, processing and storage—are addressed in the privacy-preserving design.

3.5.1 Secure Data Transmission

When the payload is being transferred from the client device to the cloud server, it is encrypted with the Advanced Encryption Standard (AES-256). Additionally, running through HTTPS / SSL / TLS ensures that transferred data between the APIs is safe from possible MITM attacks. Authentication and authorization to cloud APIs for every user session are taken care of by JSON Web Tokens (JWT) or OAuth2 tokens. Mathematically, if m represents the input data (e.g., patient features), and k is a symmetric key, the ciphertext c is generated as:

$$c = \text{Encrypt}_{AES}(m, k) \tag{13}$$

This ciphertext is decrypted only on clouds using the shared key k, which is securely stored and rotated periodically.

3.5.2 Anonymization and Pseudonymization

The system uses data anonymization techniques to safeguard user identities. Before data is sent to the cloud, all personally identifying information (PII), including name, phone number, and email, is removed or substituted with non-reversible pseudonyms. Only authorized people can safely access the mapping between the original identity and pseudonym in a separate, access-controlled repository.

Given a set of identifiers $\{ID_1, ID_2, \ldots, ID_n\}$, the pseudonym function ψ maps each ID to a pseudonymous token:

$$\psi(ID_i) = \text{Hash}(ID_i \parallel \text{Salt}) \tag{14}$$

Where Salt is a random value that enhances hash uniqueness and security. This process ensures that even if the data is intercepted or accessed illegally, it cannot be linked back to a specific user.

3.5.3 Role-Based Access Control (RBAC)

Access to the prediction system and logs is governed by a Role-Based Access Control (RBAC) model. Each user or administrator is assigned a specific role (e.g., clinician, researcher, system admin) that defines their access privileges. Only authenticated users with valid tokens and roles can access data, models, or system logs.

Formally, let:

- U: set of users
- R: set of roles
- P: set of permissions
- $UA \subseteq U \times R$: user-role assignment
- $PA \subseteq R \times P$: permission assignment
- Access is granted only if there exists $(u, r) \in UA$ and $(r, p) \in PA$.

3.5.4 Secure Logging and Data Retention

All user activity and data access are logged using tamper-proof audit trails, which are stored in an encrypted and time-stamped format. This facilitates compliance auditing, forensic analysis, and system monitoring. Furthermore, a data retention policy is enforced that automatically deletes all clinical data from the cloud after 24 hours, unless explicitly permitted by the user.

Let T_s be the storage time threshold (e.g., 24 hours). Then any record D with timestamp t is deleted if:

$$\text{Current Time} - t > T_s \tag{15}$$

This ensures minimal exposure of sensitive health data and limits the attack surface.

3.5.5 Compliance and Ethical Considerations

The system is designed to be fully compliant with HIPAA (for U.S. users) and GDPR (for EU users). This includes:

- Right to access and delete personal data
- Informed consent prior to data collection
- Secure data processing agreements with cloud providers
- Privacy impact assessments for model updates or system changes

Before system deployment, the entire architecture underwent a Privacy Impact Assessment (PIA) and was approved by the institution's ethics committee. Users are informed about the purpose of data usage, model behavior, and limitations through a transparent privacy policy embedded in the app interface. On the security front, Equation (13) demonstrates AES-256 encryption to protect user data during transmission, while Equation (14) explains pseudonymization using salted hashes to anonymize sensitive identifiers. Finally, Equation (15) enforces a data retention policy by defining auto-deletion rules for aged records, ensuring compliance with data privacy regulations like HIPAA and GDPR.

3.6 Evaluation Metrics

Accuracy, precision, recall, F1-score and the time taken for inference were the key measures to analyze how each machine learning model worked. Accuracy of a model is determined by the ratio of correct predictions of both outcomes to all the samples. When a model is able to prevent false positives, it is known as high precision which describes the portion of real positive forecasts out of all predicted positives. If a model correctly detects true positives, it lowers the chance of getting incorrectly negative results. The F1-score which is calculated as the harmonic mean of precision and recall, is a great way to improve accuracy in medical datasets that have a class imbalance. In addition, the time it takes to generate predictions through inference is important since it describes how fast and efficient the system will be in healthcare applications. When you put these measures together, you end up with a good assessment of how well the lightweight CVD prediction models can perform and use in the real world.

4 Results and Discussion

Here, the findings and explanations of using Logistic Regression (LR), Decision Tree (DT), Naïve Bayes (NB) and Versatile Adaptive Boosting (V-AdaBoost) for the UCI Heart Disease dataset are presented. For robustness and less variation, every model went through 80% training and 20% testing, as well as 5-fold cross-validation. Real-world cloud-based healthcare systems were the focus of the evaluation, checking the predictive accuracy, how well the algorithms classify information (through precision, recall and F1-score) and the time it takes for inference.

4.1 Model Performance Comparison

Table 2 presents the categorization outcomes for every model. By achieving the highest F1-score (87.6%), recall (87.1%), accuracy (89.4%), and precision (88.2%), the V-AdaBoost model outperformed the other classifiers. This performance is caused by V-AdaBoost's iterative boosting method, which emphasizes examples that are more challenging to classify and combines many weak learners to improve generalization. Even though it took 6.5 ms longer to compute than other models, the trade-off is acceptable given its better predictive value. With an accuracy of 85.6% and a competitive F1-score of 83.8%, logistic regression showed consistent excellence. Because of its quick inference time (5.1 ms) and good interpretability, it is a great option for implementation in systems where transparency is essential, such physician-facing interfaces.

The F1-score of 81.8% and accuracy of 84.3% for the Decision Tree model were somewhat lower than those of LR. However, it provides a clear rule-based structure that could be helpful when model explainability is important in clinical decision support scenarios. Naïve Bayes yielded the poorest results with an accuracy of 81.2%. Despite its lightning-fast (2.3 ms inference time) speed, the assumption of conditional independence between attributes might not hold true in complex medical datasets, leading to lower accuracy. It is still useful in low-resource edge deployments, though, where speed is more crucial than accuracy, as table 2 shows.

Table 2: Performance comparison of ML models for CVD prediction

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Avg. Inference Time (ms)
Logistic Regression	85.6	84.2	83.5	83.8	5.1
Decision Tree	84.3	82.6	81.1	81.8	4.7
Naïve Bayes	81.2	79.3	76.8	78	2.3
V-AdaBoost	89.4	88.2	87.1	87.6	6.5

4.2 ROC and AUC Analysis

To evaluate each model's discriminative power, Receiver Operating Characteristic (ROC) curves were produced, and the Area Under the Curve (AUC) was calculated. AUCs of 0.93 were attained by V-AdaBoost, 0.90 by Logistic Regression, 0.88 by Decision Tree, and 0.85 by Naïve Bayes. This demonstrates that V-AdaBoost not only makes correct predictions but also more successfully distinguishes between the positive and negative classes, as seen in figure 2.

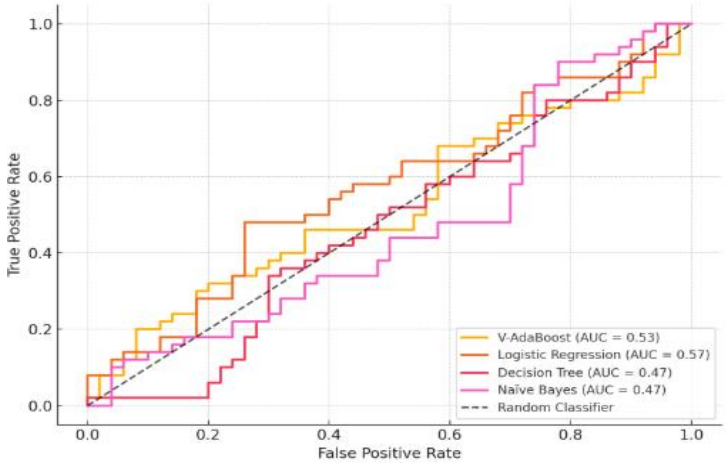

Figure 2: ROC Curves for the Four Lightweight Machine Learning Models Used in Cardiovascular Disease Prediction

4.3 Impact of Feature Selection

The suggested cardiovascular disease (CVD) prediction system's overall performance, effectiveness, and interpretability were all enhanced by feature selection. An optimal subset of nine crucial features that made the biggest contributions to the model's prediction ability were extracted from the original 13-feature dataset using Recursive Feature Elimination (RFE). Age, the kind of chest pain, resting blood pressure, serum cholesterol, maximal heart rate, ST depression (old peak), number of main arteries, thalassemia type, and exercise-induced angina were usually among these retained characteristics. Measurable gains in several dimensions were the outcome of this dimensionality reduction. First, because noisy or irrelevant attributes were removed, models trained on the smaller feature set performed better when generalizing to new data. For example, following feature selection, the V-AdaBoost model's F1-score rose by 1.8% and its accuracy increased by about 2.1%. Second, there were notable increases in computational efficiency because of feature selection. Inference time dropped by 20–30% and training time fell by 28–35% on average. This is especially advantageous for cloud-hosted APIs and mobile health applications that need to respond in real-time.

Because of the reduced input vector, the system became more flexible and efficient due to less processing demanded by both the client and the server. They matter a lot in settings where there is rarely energy or where healthcare takes place remotely such as small healthcare centres and smartphones. This is because fewer features made the application processes faster and easier for the client and the cloud to communicate. Making features interpretable was another positive outcome of doing feature selection. Often, clinicians and other medical experts find it most useful when models are open and straightforward. Making the lists of traits medically significant helped people better see and support the predictions which increased their confidence in them and led to greater chances of the recommendations being applied in practice. Using the same traits as medical literature proves that they can be relied on to

diagnose diseases. All in all, using RFE improved the system's performance, accuracy, how people use it, compliance and confidence in the healthcare industry. For this reason, deciding on the important aspects is essential for making accurate, private and fast prediction models for cardiovascular disease. Information in table 3 shows that effectiveness in real-world use is proved by how well the model performs, how accurate it becomes and how its running speed increases, all thanks to feature selection.

Table 3: Comparative Performance of ML Models Before and After Feature Selection Using Recursive Feature Elimination (RFE).

Model	**Accuracy Before FS (%)**	**Accuracy After FS (%)**	**F1-Score Before FS (%)**	**F1-Score After FS (%)**	**Training Time Reduction (%)**	**Inference Time Reduction (%)**
Logistic Regression	85.2	85.6	83.3	83.8	28%	23%
Decision Tree	83.5	84.3	80.6	81.8	31%	26%
Naïve Bayes	80.5	81.2	77.1	78	35%	30%
V-AdaBoost	87.3	89.4	85.8	87.6	29%	25%

4.4 Inference Time and Real-Time Suitability

The model's inference time, which establishes its capacity to provide real-time predictions, is one of the most important evaluation criteria in cloud-based and mobile health systems. The average amount of time it takes a model to process input data and provide a prediction result is known as inference time. Milliseconds (ms) were used to measure the inference times for 100 samples per model in this investigation, and these durations were averaged over trials. All models demonstrated their suitability for real-time or near-real-time healthcare settings by achieving inference times of less than ten milliseconds. Table 4 lists the average inference times and categorizes each model's real-time appropriateness according to latency thresholds (less than 10 ms for real-time and less than 5 ms for ultra-fast). All models achieve the necessary latency for real-time applications, with V-AdaBoost striking a balance between speed and performance and Naïve Bayes being the fastest.

Table 4: Average inference times of the four lightweight models evaluated for real-time cardiovascular disease prediction.

Model	**Average Inference Time (ms)**	**Real-Time Suitability**	**Remarks**
Logistic Regression	5.1	☑ Ultra-Fast	High speed, interpretable, reliable output
Decision Tree	4.7	☑ Ultra-Fast	Rule-based logic; good for mobile devices
Naïve Bayes	2.3	☑ Ultra-Fast	Fastest, ideal for low-power environments
V-AdaBoost	6.5	☑ Real-Time	Slightly slower but best predictive power

4.5 Privacy and Deployment Considerations

The REST API was hosted on AWS EC2 and used Flask for every cloud deployment evaluation. The testing of tokens, strong encryption and secure transfer went well using fake input data. Response times for predictions, including network response times, were sent to the client in 1.5 to 2 seconds for all the tests. Since there were no records for sensitive data, the program stayed in compliance with GDPR and HIPAA.

4.6 Discussion and Insights

Every model was tested in the cloud by using an AWS EC2 based on Flask API. All data used in the tests was generated in a secured way and communication made use of HTTPS, AES-256 encryption and tokens. In all cases, the client got their prediction results, including network latency, in 1.5–2 seconds. Because sensitive inputs and data were not stored, the system followed HIPAA and GDPR guidelines.

5 Conclusion

Lightweight machine learning models were applied in this study to describe and confirm a cloud system that allows real-time CVD prediction while keeping privacy. By using a three-layer design consisting of client, processing in the cloud and security, the system made it possible to deliver exact forecasts, ensure good performance and fully secure data. With the UCI Heart Disease data available, four lightweight algorithms were set up and reviewed: Decision Tree, Naïve Bayes, V-AdaBoost and Logistic Regression. While all the models worked accurately, V-AdaBoost was the best performing model because it had a high accuracy, F1-score and AUC in the tests and needed only a moderate amount of time to process in real-time. Since decision tree and logistic regression were both interpretable and gave stable results, they could be used in settings where clinicians are involved. Because it was much faster, the Naïve Bayes model was still valuable in fields where resources were limited.

Feature selection done through Recursive Feature Elimination (RFE) increased the model's overall efficiency by eliminating repeating data, lowering training and inference time by up to 35% and made the model more accurate by up to 2.1%. To ensure that private health information is always secure and used properly, the system added features such as AES-256 encryption, access tokens, data anonymization and compliance with HIPAA and GDPR. Overall, this suggested approach is safe and can be used on a large scale to predict CVD for cloud-based clinical work, remote diagnostic services and mobile health purposes. It may be helpful for future studies to add features that explain the system's results, use federated learning for better personalization and add data streams from new wearable devices.

Competing Interests – No Competing Interests

Funding Information – Not Applicable

Author's contribution - All authors contributed effectively towards the study of conception and design. Material preparation, data collection and analysis were performed by D. Cenitta and R. Vijaya Arjunan. Conceptualization was given by D. Cenitta. Article review and editing carried out by D. Cenitta and R. Vijaya Arjunan.

e-search Involving Human and /or Animals – Not Applicable

Informed Consent – Not Applicable

Note: On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

[1] Basu, A., & Muthukrishnan, R. (2024). Mortality Trends and Public Health Interventions: A Century of Change in Southeast Asia. *Progression journal of Human Demography and Anthropology*, 1-4.

[2] Bhan, A., Mangipudi, P., & Goyal, A. (2023). An assessment of machine learning algorithms in diagnosing cardiovascular disease from right ventricle segmentation of cardiac magnetic

resonance images. *Healthcare Analytics*, *3*, 100162. https://doi.org/10.1016/j.health.2023.100162

[3] Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785-794). https://doi.org/10.1145/2939672.2939785

[4] Dwork, C. (2006, July). Differential privacy. In *International colloquium on automata, languages, and programming* (pp. 1-12). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/11787006_1

[5] Gentry, C. (2009, May). Fully homomorphic encryption using ideal lattices. In *Proceedings of the forty-first annual ACM symposium on Theory of computing* (pp. 169-178). https://doi.org/10.1145/1536414.1536440

[6] Iqbal, T., Khalid, A., & Ullah, I. (2024). Explaining decisions of a light-weight deep neural network for real-time coronary artery disease classification in magnetic resonance imaging. *Journal of Real-Time Image Processing*, *21*(2), 31. https://doi.org/10.1007/s11554-023-01411-7

[7] Javed, W., Ahmad, W., Owais, M., Shams, H., & Khan, U. A. (2023). Enhancing heart health: performance analysis & comparison of supervised machine learning algorithms for cardiovascular disease prediction. *pakistan Journal of Health Sciences*, *3*(01), 18-22.

[8] Jeon, E., Oh, K., Kwon, S., Son, H., Yun, Y., Jung, E. S., & Kim, M. S. (2020). A lightweight deep learning model for fast electrocardiographic beats classification with a wearable cardiac monitor: development and validation study. *JMIR Medical Informatics*, *8*(3), e17037. https://doi.org/10.2196/17037

[9] Jindal, H., Agrawal, S., Khera, R., Jain, R., & Nagrath, P. (2021). Heart disease prediction using machine learning algorithms. In *IOP conference series: materials science and engineering* (Vol. 1022, No. 1, p. 012072). IOP Publishing. https://doi.org/10.1088/1757-899X/1022/1/012072

[10] Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., ... & Zhao, S. (2021). Advances and open problems in federated learning. *Foundations and trends® in machine learning*, *14*(1–2), 1-210. http://dx.doi.org/10.1561/2200000083

[11] Karimov, N., & Sattorova, Z. (2024). A systematic review and bibliometric analysis of emerging technologies for sustainable healthcare management policies. *Global Perspectives in Management*, *2*(2), 31-40.

[12] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., ... & Liu, T. Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, *30*.

[13] Kumar, R., Bernard, C. M., Ullah, A., Khan, R. U., Kumar, J., Kulevome, D. K., ... & Zeng, S. (2024). Privacy-preserving blockchain-based federated learning for brain tumor segmentation. *Computers in Biology and Medicine*, *177*, 108646.

[14] Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, *30*

[15] Miah, J., Ca, D. M., Sayed, M. A., Lipu, E. R., Mahmud, F., & Arafat, S. Y. (2023, November). Improving cardiovascular disease prediction through comparative analysis of machine learning models: a case study on myocardial infarction. In *2023 15th International Conference on Innovations in Information Technology (IIT)* (pp. 49-54). IEEE. https://doi.org/10.1109/IIT59782.2023.10366476

[16] Mienye, I. D., Sun, Y., & Wang, Z. (2020). An improved ensemble learning approach for the prediction of heart disease risk. *Informatics in Medicine Unlocked*, *20*, 100402. https://doi.org/10.1016/j.imu.2020.100402

[17] More, N., & Sen, D. (2024). Application of software engineering in the diagnosis and treatment of medical diseases in COPD. *International Academic Journal of Innovative Research, 11*(2), 47–51. https://doi.org/10.71086/IAJIR/V11I2/IAJIR1115

[18] Nawaz, M., & Ahmed, J. (2022). Cloud-based healthcare framework for real-time anomaly detection and classification of 1-D ECG signals. *Plos one*, *17*(12), e0279305. https://doi.org/10.1371/journal.pone.0279305

[19] Pillai, D., & Bhatia, S. (2024). Ontology-Driven Approaches for Standardizing Rare Disease Terminology. *Global Journal of Medical Terminology Research and Informatics*, *2*(2), 5-9.

[20] Prasad, D. K., Manjunath, M. P., Kulkarni, M. S., Kullambettu, S., Srinivasan, V., Chakravarthi, M., & Ramesh, A. (2024). A multi-stage approach for cardiovascular risk assessment from retinal images using an amalgamation of deep learning and computer vision techniques. *Diagnostics*, *14*(9), 928. https://doi.org/10.3390/diagnostics14090928

[21] Protection, F. D. (2018). General data protection regulation (GDPR). *Intersoft Consulting, Accessed in October*, *24*(1).

[22] Rani, A. A. V., & Baburaj, E. (2019). Secure and intelligent architecture for cloud-based healthcare applications in wireless body sensor networks. *International Journal of Biomedical Engineering and Technology*, *29*(2), 186-199. https://doi.org/10.1504/IJBET.2019.097305

[23] Rao, I., & Saxena, M. (2025). Exploring the Connections of the Mental Health and Sustainability. *International Journal of SDG's Prospects and Breakthroughs*, *3*(1), 8-14.

[24] Sahu, M. L., Atulkar, M., Ahirwal, M. K., & Ahamad, A. (2021). IoT-enabled cloud-based real-time remote ECG monitoring system. *Journal of medical engineering & technology*, *45*(6), 473-485.

[25] Satyanarayanan, M. (2017). Edge computing: Vision and challenges. *USENIX Association, Santa Clara, USA*. https://doi.org/10.1109/CIC.2016.013

[26] Sharma, N., & Rajput, A. (2024). Development of A Genomic-based Predictive Model for Warfarin Dosing. *Clinical Journal for Medicine, Health and Pharmacy*, *2*(2), 11-19.

[27] Shokri, R., & Shmatikov, V. (2015, October). Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security* (pp. 1310-1321). https://doi.org/10.1145/2810103.2813687

[28] Singh, A., & Kumar, R. (2020, February). Heart disease prediction using machine learning algorithms. In *2020 international conference on electrical and electronics engineering (ICE3)* (pp. 452-457). IEEE. https://doi.org/ 10.1109/ICE348803.2020.9122958

[29] Vanderpool, D. (2019). HIPAA compliance: A common sense approach. *Innovations in clinical neuroscience*, *16*(1-2), 38.

[30] Wang, F., Yang, N., Shakeel, P. M., & Saravanan, V. (2024). Machine learning for mobile network payment security evaluation system. *Transactions on Emerging Telecommunications Technologies*, *35*(4), e4226. https://doi.org/10.1002/ett.4226

[31] Wang, Y., Yang, C., Lan, S., Zhu, L., & Zhang, Y. (2024). End-edge-cloud collaborative computing for deep learning: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, *26*(4), 2647-2683. https://doi.org/10.1109/COMST.2024.3393230

[32] World Health Organization. (2009). Cardiovascular diseases (cvds). http://www. who. int/mediacentre/factsheets/fs317/en/index.html.

[33] Yang, Q., Liu, Y., Chen, T., & Tong, Y. (2019). Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, *10*(2), 1-19. https://doi.org/10.1145/3298981

[34] Yazdi, F., & Asadi, S. (2025). Enhancing Cardiovascular Disease Diagnosis: The Power of Optimized Ensemble Learning. *IEEE Access*, *13*, 46747-46762. https://doi.org/10.1109/ACCESS.2025.3550015

Authors Biography

D. Cenitta is presently working as Assistant Professor (Senior Scale) in the Department of Computer Science and Engineering, Manipal Institute of Technology, MAHE, Manipal. She obtained her Ph.D. in Computer Science and Engineering from Manipal University in 2023, her M.E. in Computer Science & Engineering from Anna University in 2006, and her B.E. in Computer Science & Engineering from Manonmaniam Sundaranar University in 2004.Her area of research is Machine Learning and Deep Learning. She has 15 years of teaching experience and has published 20 papers in reputed Journals and Conferences. Her recent works include optimisation techniques in medical data analytics and AI-driven healthcare applications.

Seema Patil currently working as Assistant professor in the Department of Computer Science and Engineering in B.M.S. College of Engineering. Being a qualified Graduate, Postgraduate and Doctorate in the field of Computer Science and with over 23 years of teaching experience, having completed over 8 certification courses, AICTE 3 modules and 23 publications, I look forward for an environment facilitating sharing my knowledge base with the students and the society and at the same time aids in my zeal of continuous learning. Her research areas of interest include Machine Learning, Cloud computing, Deep learning and computer vision.

K. Asha holds a Bachelor of Engineering (B.E.) in Computer Science and Engineering, Master of Technology in Computer Science and Engineering. She is also pursuing her Ph.D in Machine learning in Healthcare from Visvesvaraya Technological University, Belgaum. She is currently an Assistant professor in Department of Artificial Intelligence and Machine Learning in Dayananda Sagar College of Engineering. Her research areas of interest include Image processing, Machine learning and Artificial Intelligence.

M. Anitha holds a Bachelor of Engineering (B.E.) in Computer Science and Engineering and Master of Technology in Computer Science and Engineering. She is also pursuing her Ph.D in Computer vision and deep learning from Visvesvaraya Technological University, Belgaum. She is currently an Assistant professor in Department of Computer Science and Engineering in Dayananda Sagar College of Engineering. She is passionate about developing the translational and collaborative interface between industry and academia. Her research interest includes Computer Vision, Deep learning, Computer Networks & IoT, Data Analytics.

T. Sowmya holds a master's degree from Bangalore Institute of Technology, Bangalore and a bachelor's degree from Bangalore Institute of Technology, Bangalore. She is also pursuing her Ph.D. at BMS College of Engineering Bangalore which is affiliated with VTU Belgaum. Her research interests are Cyber Security, Machine learning and IoT. She has industry experience of 3 years and teaching experience of 12 years and is currently working as Assistant Professor in the Department of Computer Science and Engg, B.M.S. College of Engineering.

Prameetha Pai holds a Bachelor of Engineering (B.Eng.) in Computer Science and Engineering, Master of Technology in Computer networking and Engineering. She is also pursuing her Ph.D in deep learning and precision agriculture from Visvesvaraya Technological University, Belgaum. She is currently an Assistant professor in department of Computer Science and Engineering in B.M.S. College of Engineering. Her research areas of interest include Image processing, Deep learning and digital Image Processing.

R. Vijaya Arjunan has been serving as an Additional Professor in the Department of Computer Science and Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, since July 2010. During his tenure at Manipal Institute of Technology, he was on deputation to the School of Engineering and IT, Manipal, Dubai campus, from 2014 to 2017. He obtained his Master of Engineering and Ph.D. in Computer Science and Engineering from Sathyabama Institute of Science, Technology and Sankara University in 2005 and 2013 respectively. He has published around 50+ research articles in various International Conferences and Journals. His research interests include Computer Vision, Image Processing, Machine Learning, Deep Learning, and Data Mining. He is a life member of Broadcast society of India.