An Enhanced Crow Search-optimized Bi-LSTM for Heart Disease Prediction: Hybrid Feature Selection and Hyperparameter Tuning

Narayan Jee [1,*], Gesu Thakur [1], and Sumit Kumar [2]

[1] College of Smart Computing, COER University, Roorkee, Uttarakhand, India
[2] Department of CSE, Sharda University, Greater Noida, India
Email: narayan.btech@gmail.com (N.J.); drgesuthakur@gmail.com (G.T.); mail2dr.sumit@gmail.com (S.K.)
*Corresponding author

Manuscript received November 23, 2025; revised January 26, 2026; accepted April 21, 2026; published August 7, 2026

Abstract*—Accurate and early prediction of Cardiovascular Disease (CVD) plays an important role in improving patient survival and enabling timely preventive or therapeutic interventions. In this study, we propose an enhanced Crow Search Algorithm (eCSA) that integrates feature selection and hyperparameter optimization into a unified framework to improve the performance of a Bidirectional Long Short-Term Memory (Bi-LSTM) based heart disease prediction network. By employing a hybrid binary-continuous search strategy, the proposed eCSA jointly identifies discriminative features and optimizes Bi-LSTM hyperparameters, resulting in improved predictive performance and enhanced feature-level interpretability. Experimental evaluations were conducted on two benchmark datasets, namely the Cleveland Heart Disease dataset and the Framingham Heart Study dataset, under multiple train-test split conditions (80:20 and 70:30). Comparative analysis against state-of-the-art approaches, including Stochastic Configuration Network–Deep Bidirectional Long Short-Term Memory (SCN-Deep Bi-LSTM), Fitness-based Horse Optimization–Bidirectional Long Short-Term Memory (FHO-Bi-LSTM), Quantum Hybrid Deep Network (QHDN), and Convolutional Transformer Network (CTN-Trans), demonstrated that the proposed eCSA-BiLSTM model consistently outperformed competing methods across multiple evaluation metrics. Performance was assessed using accuracy, macro-averaged F1-Score (Macro-F1), precision, recall, Matthews Correlation Coefficient (MCC), and Area Under the Receiver Operating Characteristic Curve (AUC-ROC). The proposed model achieved an accuracy of up to 0.95 on the Cleveland dataset and 0.91 on the Framingham dataset, while maintaining stable and consistent performance across stratified 5-fold cross-validation and repeated train–test split experiments.

***Keywords*—enhanced Crow Search Algorithm (CSA), heart disease prediction, hyperparameter tuning**

I. Introduction

Cardiovascular Diseases (CVD) remain a leading cause of death throughout the world, with an estimated 17.9 million deaths per year, contributing to an alarmingly high proportion of global mortality. Accurate and timely diagnosis is crucial for effective clinical intervention and improved patient outcomes. However, conventional diagnostic methods that utilize classical statistical models often struggle with high-dimensional medical data and nonlinear relationships between predictors and disease.

Recent developments in Deep Learning (DL)—and more specifically, Bidirectional Long Short-Term Memory (Bi-LSTM) networks—have greatly improved modeling structures for sequential and temporal patterns in patient-oriented input data. Bidirectional Long Short-Term Memory (Bi-LSTM) networks are traditionally designed for sequential data; however, recent studies have demonstrated that recurrent architectures can also be adapted to tabular clinical datasets by modeling structured inter-feature relationships [1]. In such settings, features are treated as an ordered input sequence, enabling the network to capture higher-order dependencies and nonlinear interactions among clinical variables. This perspective allows Bi-LSTM to function as a feature interaction learner rather than a temporal sequence model [2]. However, these models usually require careful hyperparameter tuning (e.g., learning rate value, number of layers, weight initialization, and bias values), which are commonly performed by computationally expensive techniques such as grid search or random search [3].

In parallel, selection of features is crucial to reduce the complexity and enhance interpretability of the models by excluding irrelevant or redundant factors [4]. Feature selection in combination with hyperparameter optimization may result in more parsimonious, interpretable, and accurate predictive models.

Inspired by the collective behavior in nature, swarm intelligence has recently emerged as a formidable arsenal for optimization problems. Algorithms such as Particle Swarm Optimization and Genetic Algorithms have shown great success in parameter tuning and feature selection [5]. Among them, the Crow Search Algorithm (CSA) exhibits a strong balance between exploration and exploitation, since it is inspired by the hoarding and memory of crows [6].

Variants of CSA have been adapted for discrete tasks and hybrid optimization:

- Zamani and Nadimi-Shahraki [6] introduced an Evolutionary Crow Search Algorithm (ECSA) to optimize neural network hyperparameters for chronic disease diagnosis.
- Sinha *et al.* [7] applied a Crow Search-based feature selection and hyperparameter scheme in an online Hybrid Machine Learning (ML) classifier for heart disease prediction, demonstrating improved accuracy by shrinking data noise.
- Similarly, Predator Crow Optimization (PCO) combining elements of crow behavior with predator-prey characteristics—has been employed for deep neural network weight optimization for cardiovascular diagnosis and Electrocardiogram (ECG) using a solution that is both fast and accurate [8].

- In addition, a framework that integrated Predator Crow Search Optimization and Explainable AI (XAI) techniques was developed, achieving high accurate classification (≈99.7% accuracy) along with improved model interpretability [9].

Although this is a significant step forward, most existing approaches still treat feature selection and hyperparameter tuning as sequential stages in isolation, potentially overlooking complementary interactions between input dimensionality and model parameterization. In addition, to the best of our knowledge, swarm-based joint optimization has not been integrated with cutting-edge architecture (e.g. Bi-LSTM) in this particular cardiovascular field.

To fill these voids, this research introduces an improved Crow Search-based joint optimization framework, which jointly targets:

(1) Feature Selection: Identifying the most informative clinical attributes.
(2) Hyperparameter Tuning: Optimizing critical Bi-LSTM parameters, including weight and bias initialization and learning rates.

Key contributions include:

- A unified swarm-based optimization strategy that utilizes a modified CSA to calibrate feature subsets and model parameters in a single process.
- Enhanced model feature level interpretability and classification performance—improvements in accuracy, sensitivity, and F1-Score are demonstrated.
- Extensive benchmarking against standard Bi-LSTM models and baseline tuning methods (e.g., grid/random search, sequential feature selection) on standard heart disease datasets.
- A novel exploration of hybrid swarm intelligence in DL-based clinical decision support systems, reinforcing the potential of such methods in cardiovascular healthcare.

The rest of the paper is organized as follows: Section II reviews related to works on feature selection, hyperparameter optimization, and swarm intelligence in healthcare. Section III details the proposed methodology, including data set setup, preprocessing, enhanced CSA, and Bi-LSTM architecture. Section IV presents experimental results and comparative analysis. Section V concludes with future research directions.

II. Literature Review

Cardiovascular Disease (CVD) prediction from clinical tabular features has long served as a benchmark problem for machine-learning pipelines balancing accuracy, interpretability, and computational cost. Early work relied on traditional classifiers trained with UCI "Cleveland"-style tabular datasets (13 to 76 variables, mixed types) and emphasized careful preprocessing and Feature Selection (FS) (to reduce redundancy and collinearity). The UCI Heart Disease repository remains one of the most widely used benchmarks for evaluating classification models in cardiovascular prediction [10].

A. Deep Learning for Cardiac Prediction and Sequence Signals

Sequence-aware architectures, particularly LSTMs [11], and their bidirectional counterparts—have performed well on cardiac signals (e.g., ECG) and more recently, on tabular or fused multimodal inputs with the rise of deep learning maturity. Bi-LSTMs can also model temporal dependencies from both forward and backward directions, which is useful for accommodating variations in rhythmic patterns and longer span context. Recent works combine Bi-LSTMs with CNN front ends or attention and achieve significant improvements in Massachusetts Institute of Technology-Beth Israel Hospital (MIT-BIH) Arrhythmia Database benchmark and screening tasks [11–13]. For instance, Hassan *et al.* [12] fuse CNN and Bi-LSTM for arrhythmia annotation, which consistently outperform traditional pipelines. In the heart-disease (tabular) setting, Veerabaku *et al.* [11] use an "intelligent Bi-LSTM" with architecture optimization to predict heart disease from clinical features, underscoring the importance of principled tuning and input selection. In addition to temporal applications, recent research has explored the adaptation of sequence-based architectures for tabular data by imposing a deterministic ordering over input features. This approach enables recurrent networks to learn complex inter-feature dependencies that are not explicitly captured by conventional machine learning models. Such adaptations are particularly relevant in clinical datasets, where interactions among risk factors may exhibit nonlinear and context-dependent behavior. Therefore, sequence models like Bi-LSTM can be interpreted as structured feature interaction models when applied to tabular inputs.

B. Feature Selection: Filters, Wrappers, Embedded and Evolutionary Methods

FS improves generalization, reduces cost, and enhances model transparency. Surveys consolidate FS into 3 broad families—filter (model-agnostic), wrapper (search guided by a learner), and embedded (selection occurs during training)—each with trade-offs in compute and stability. Evolutionary Computation (EC)-based wrappers are especially attractive for high-dimensional biomedical data: they search large, rugged spaces, handle mixed data types, and directly optimize downstream metrics (e.g., F1-Score sensitivity). Empirically, across heart-disease datasets, wrapper and EC-based FS tend to boost sensitivity/specificity versus filter-only baselines, while also reducing the selected subset size.

C. Meta-Heuristic FS and Hybrid Pipelines in Heart-Disease Prediction

A substantial body of research has employed nature-inspired meta-heuristic optimization techniques to improve Feature Selection (FS) and classifier parameter tuning for heart disease prediction. Common approaches include the Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Grey Wolf Optimizer (GWO), Whale Optimization Algorithm (WOA), and Ant Colony Optimization (ACO). These optimization methods are frequently integrated with machine learning and deep learning classifiers to enhance predictive performance and reduce feature redundancy.

Representative studies have demonstrated the effectiveness of Genetic Algorithm-based feature selection combined with Support Vector Machine (SVM) and Artificial Neural Network (ANN) classifiers, Particle Swarm Optimization-driven subset selection methods, and Grey Wolf Optimizer-assisted ensemble learning frameworks. These approaches consistently report improved classification

accuracy on benchmark datasets, including the Cleveland Heart Disease and Statlog Heart Disease datasets.

More recent investigations have explored stacked learning architectures integrated with meta-heuristic search strategies and reported further gains in predictive performance, particularly in terms of macro F1-Score (F1) and Area Under the Receiver Operating Characteristic Curve (AUC-ROC). These findings suggest that optimization and search mechanisms can have an influence comparable to, and in some cases greater than, the selection of the underlying classifier architecture itself.

D. Crow Search Algorithm (CSA): Foundations and Advances

CSA is a swarm-intelligence optimizer inspired by crows' food-hiding and pilfering strategies. Askarzadeh's original formulation features a simple position update with only two key parameter flight length and awareness probability—making it lightweight yet competitive across constrained engineering benchmarks [14]. Subsequent surveys document rapid adoption of CSA across domains and classify dozens of improved or hybridized variants designed to enhance exploration/exploitation balance [15]. In structural optimization and beyond, CSA's few hyperparameters and memory mechanism (personal best) make it an appealing choice when objective evaluations are costly or noisy [14, 15].

E. CSA for Feature Selection (Binary, Chaotic, and Enhanced Variants)

To apply CSA to FS, researchers typically use binary encodings (presence/absence of features) and design fitness functions that combine classifier loss with subset cardinality. Sayed *et al.* [16] introduced a chaotic CSA that perturbs trajectories to avoid premature convergence, reporting smaller subsets and better accuracy on standard FS benchmarks. Ouadfel and Elaziz [17] proposed an "Enhanced CSA" with modified position updates and adaptive parameterization, demonstrating gains across high-dimensional FS tasks. In more recent chemometrics and big-data contexts, Al-Thanoon *et al.* [18] adapted CSA for scalable FS and showed improved classification in multi-class settings. These works collectively suggest that CSA variants can act as strong FS wrappers, especially when the base learner is sensitive to redundant or noisy inputs [16–18].

F. CSA Hybrids and Memory-Enhanced Designs

A parallel line improves CSA via hybridization (e.g., Grey Wolf + CSA, Sine-Cosine + CSA) or memory-augmented updates that stabilize convergence. Memory-based hybrid CSA has been shown to outperform vanilla CSA on a range of numeric and constrained problems by preserving high-quality regions while injecting exploratory moves [19]. Such designs are particularly relevant to FS with rugged objectives (e.g., F1-Score under cross-validation), where balance between diversification and intensification is crucial [15, 19].

G. Hyper-Parameter Optimization (HPO): from Random Search to Bayesian Optimization

Even with a strong FS strategy, modern learners (including Bi-LSTMs) require careful HPO. Foundational results show random search is typically more efficient than grid search in high-dimensional spaces because performance is often sensitive to only a few dimensions. Bayesian Optimization (BO) with Gaussian-process surrogates further reduces trials by trading off exploration vs. exploitation via acquisition functions; Snoek *et al.* [20] demonstrated BO can match or surpass expert-level tuning across diverse ML models. These results motivate coupling FS with principled HPO in a single pipeline that jointly optimizes the input subspace and model configuration [21].

H. CSA for Neural Network Weight/Bias Tuning and Medical Prediction

Beyond Feature Selection (FS), the Crow Search Algorithm (CSA) has also been employed to initialize and directly optimize neural network weights and biases to improve learning efficiency and predictive performance. Amor *et al.* [22] proposed a Crow Search Algorithm-optimized Multi-Layer Perceptron (MLP-CSA), in which CSA was used to optimize the parameter space of the neural network during training. Their results demonstrated improved regression accuracy compared with Particle Swarm Optimization (PSO)-based training, Genetic Algorithm (GA)-based optimization, and conventional backpropagation approaches. These findings indicate that CSA can function as an effective weight-space optimization strategy and highlight its capability to enhance convergence behavior and model generalization beyond conventional gradient-based learning methods. In healthcare, Alqurashi *et al.* [8] proposed a Predator Crow Optimization (PCO)-tuned deep neural network for Crow Search Algorithm (CSA)-based optimization for tuning deep neural networks in Cardiovascular Disease (CVD) prediction under Internet of Things (IoT)-enabled healthcare environments. These studies demonstrated the feasibility of crow-inspired Hyperparameter Optimization (HPO) for clinical risk modelling, showing that intelligent search mechanisms can effectively improve predictive performance, model adaptability, and decision support in data-driven healthcare applications.

These studies—while not focused on Bi-LSTM—strengthen the case for crow-based optimization to (i) pick informative subsets of features and (ii) tune deep nets' continuous hyperparameters. Table 1 shows the key findings of the existing state-of-the-art methods.

The research gaps include: (1) FS remains central for tabular heart-disease prediction and for trimming inputs before or alongside deep models [23, 24]. (2) Meta-heuristic FS—especially EC-based wrappers—show robust gains on Cleveland-like datasets [18, 25, 26]. (3) CSA and its variants are competitive, with multiple reports of improved convergence and better FS subsets vs. baselines [16–18]. (4) In parallel, state-of-the-art HPO (random/BO) is known to be critical for deep learners [20, 21]. The present work targets precisely this gap by proposing an enhanced CSA that integrates binary FS and continuous HPO for Bi-LSTM and by validating the approach across benchmark heart datasets with sensitivity/F1 as primary endpoints.

Table 1. Key findings of the existing state-of-the-art methods

Authors & Year	Method/ Algorithm	Application/ Dataset	Key Findings
Askarzadeh (2016) [14]	Crow Search Algorithm (CSA)	Constrained engineering benchmarks	Introduced CSA; robust performance with simple design.
Meraihi *et al.* (2021) [15]	Survey of CSA and variants	Various optimization and ML domains	Comprehensive taxonomy of CSA applications and improvements.
Sayed *et al.* (2019) [16]	Chaotic CSA for feature selection	FS benchmarks with classifiers	Achieved smaller subsets and improved accuracy.
Ouadfel & Elaziz (2020) [17]	Enhanced CSA for feature selection	High-dimensional datasets	Improved exploration/exploitation balance; better FS accuracy.
Al-Thanoon *et al.* (2021) [18]	CSA for big-data classification FS	Chemometrics and large-scale data	Efficient FS; reduced dimensionality with improved classification.
Chaudhuri & Sahu (2021) [25]	Hybrid metaheuristic FS + ML classifiers	Heart disease datasets (Cleveland, Statlog)	Increased accuracy and sensitivity vs baselines.
Veerabaku *et al.* (2023) [11]	Intelligent Bi-LSTM with architecture tuning	Clinical heart disease dataset	Bi-LSTM architecture optimization improved prediction accuracy.
Hassan *et al.* (2022) [12]	CNN + Bi-LSTM	MIT-BIH arrhythmia dataset	Outperformed classical methods in ECG classification.
Snoek *et al.* (2012) [20]	Bayesian optimization for HPO	Neural networks, SVMs, ML algorithms	Demonstrated BO efficiency for hyperparameter tuning.
Alqurashi *et al.* (2023) [8]	Predator Crow Optimization+DNN	IoT cardiovascular dataset	Enhanced accuracy and reduced computation time for CVD prediction.

III. Methodology

This section formalizes enhanced optimization of (i) a binary Feature-Selection (FS) vector and (ii) a continuous/discrete hyper-parameter vector for a Bi-LSTM classifier using an enhanced Crow Search Algorithm (eCSA). It specifies the learning problem and search-space encoding, then presents the fitness function, the mathematical formulation of eCSA and the pseudocode.

A. Problem Definition

Let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ be a labelled heart-disease dataset, where $x_i \in \mathbb{R}^d$ represents the clinical feature vector and $y_i \in \{0, 1\}$ denotes the class label.

We aim to jointly optimize:

- A feature subset $S \subseteq \{1,2, \dots, d\}$, and
- A Bi-LSTM classifier $f_\theta(\cdot)$ parameterized by hyperparameters θ,

such that predictive performance is maximized while maintaining compactness of the selected feature set.

This is formulated as minimizing the scalarized objective:

$$\mathcal{J}(S, \theta) = -\mathcal{M}(S, \theta) + \lambda \frac{|S|}{d}$$

where:

- $\mathcal{M}(S, \theta)$ is the validation metric (macro-F1 averaged over K-fold cross-validation),
- $|S|$ is the number of selected features,
- d is the total number of features,
- λ controls the trade-off between predictive accuracy and subset compactness.

B. Bi-LSTM Classifier and Preprocessing

Although the employed datasets consist of tabular clinical features rather than temporal sequences, the Bi-LSTM architecture is utilized in this work as a structured feature interaction model. Specifically, each feature dimension is mapped to an ordered input step, allowing the bidirectional recurrent mechanism to capture dependencies across feature dimensions. This formulation does not assume any temporal relationship among variables; instead, it leverages the representational capacity of recurrent networks to model complex, nonlinear interactions between clinical attributes. Such a design enables the network to learn contextual feature relationships that may not be effectively captured by traditional classifiers.

1) Preprocessing

All preprocessing operations were performed strictly within each training fold during cross-validation. Missing-value imputation (median strategy) and feature scaling were computed exclusively on the training subset of each fold and subsequently applied to the corresponding validation/test subset. At no stage were statistics derived from the test data used during training. This fold-wise protocol prevents data leakage and ensures unbiased performance estimation. Fig. 1 illustrates a conceptual workflow; in practice, preprocessing is embedded within the cross-validation loop and executed independently for each fold to eliminate any possibility of information leakage.

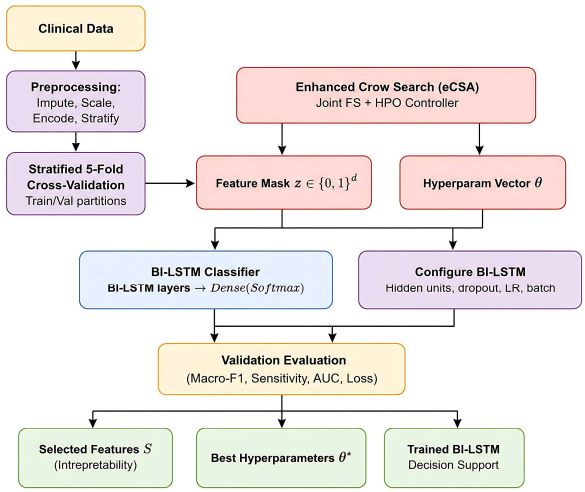

Fig. 1. Flow of eCSA.

2) Architecture

The base model is a 1–2-layer Bi-LSTM with hidden size H, dropout p between recurrent/fully connected layers, and a final dense layer with softmax. The Bi-LSTM model is trained using categorical cross-entropy loss with class weights computed from inverse class frequencies to mitigate class imbalance effects, particularly in the Framingham dataset. This weighting mechanism penalizes minority-class misclassification more strongly and improves balanced

evaluation metrics such as macro-F1 and Matthews Correlation Coefficient (MCC). Early stopping is applied based on validation loss to prevent overfitting. The model is optimized using the Adam optimizer with learning rate η. Some hyperparameters are continuous (η, p, λ_{l2}), while others are integer-valued (H and batch size). Fig. 1 illustrates the overall workflow of the proposed eCSA framework. The complete procedure of the proposed Enhanced Crow Search-based joint feature selection and hyperparameter optimization framework is presented in Algorithm 1.

Algorithm 1. Enhanced Crow Search for Joint FS + HPO (Bi-LSTM)

Input: Dataset D, population size N, iterations T, fold count K
Bounds L, U for each decision variable (binary pre-activations and hyperparams)
Weights (w_1, w_2), elite size K_{elite} , schedules $(AP_t, FL_t, \eta_t, \sigma_t)$,
decode map $\mathfrak{D}(\cdot)$, penalty settings

Initialize positions $\{x_i^0\}_{\{i=1\}}^N \sim Uniform(L, U)$
For each i: decode $(z_i, \theta_i) = \mathfrak{D}(x_i^0)$
Evaluate $J_i = J(z_i, \theta_i)$ via K-fold CV
Set memory $m_i^0 = x_i^0$
Set global best $g^0 = argmin_i J(z_i, \theta_i)$
for $t = 0$ to $T - 1$ do
Construct elite set E^t as K_{elite} best memories
for $i = 1\ to\ N$ do
Select leader j:
With prob p_{elite} choose $j \in E^t$ uniformly, else $j \in \{1..N\}$
Draw $u \sim Uniform(0,1)$
if $u \geq AP^t$ **then**
$x_i^t + r_i^t \odot FL^t \odot (m_j^t - x_i^t)$, $r \sim Uniform(0,1)^D$
else
$x\_i \leftarrow \alpha\, L + (1-\alpha)(U + L - x_i^t) + \delta^t, \alpha \sim U(0,1), \delta^t$
end if
$x_i \leftarrow clip\_to\_bounds(x_i + \eta^t * \varepsilon, \varepsilon \sim N(0, \sigma_t^2 I))$
$(z_i, \theta_i) \leftarrow \mathfrak{D}(x_i)$
if $|S_i| > m_{min}$ **then**
$J_i \leftarrow J(z_i, \theta_i)$ via K-fold CV
end if
if $J_i < J(m_i^t)$ then $m_i^{t+1} \leftarrow x_i$ **else** $m_i^{t+1} \leftarrow m_i^t$
end for
Update global best $g^{t+1} = argmin_i J(m_i^{t+1})$
Update schedules $AP^{t+1}, FL^{t+1}, \eta^{t+1}, \sigma^{t+1}$
end for
$(z*, \theta*) \leftarrow decode(g^T)$
Train final Bi-LSTM on full training data with $(z*, \theta*)$ and return model

Output: Best $(z*, \theta*)$ and trained Bi-LSTM $f_{\{\theta*\}}$

The proposed Bi-LSTM architecture was optimized using the enhanced Crow Search Algorithm (eCSA), which explored a candidate space comprising one to two Bi-LSTM layers with varying hidden units and dropout configurations. While this range defines the search space, the final architecture used for all reported experiments corresponds to the optimal configuration selected by eCSA.

After optimization, the final configuration selected by eCSA consists of two bidirectional LSTM layers with 128 hidden units each, followed by a dropout layer (rate = 0.3) and a fully connected dense layer with ReLU activation, culminating in a softmax output layer for binary classification. The model is trained using the Adam optimizer (learning rate = 1×10^{-3}, batch size = 64) with categorical cross-entropy loss and early stopping (patience = 10 epochs). These parameters were fixed across all experiments to ensure consistency and reproducibility.

To avoid unstable ultra-sparse solutions that may arise due to stochastic exploration in metaheuristic search, a mild regularization penalty is applied when the selected feature subset falls below 30% of the total feature space. Importantly, this mechanism does not enforce a strict lower bound but introduces a soft constraint within the fitness function to discourage degenerate solutions that may overfit due to extreme dimensionality reduction. In small clinical datasets such as Cleveland (13 features), excessively sparse subsets (e.g., 1–2 features) can lead to unstable cross-validation performance and inflated variance. The threshold therefore improves solution stability and reproducibility while still achieving meaningful dimensionality reduction compared to the full feature set.

While simpler classifiers offer computational efficiency, the empirical results indicate that the Bi-LSTM architecture enhances generalization by capturing structured inter-feature dependencies via bidirectional information flow across the feature space, without relying on temporal dynamics.

Although the Cleveland dataset contains a relatively small number of features (13), the proposed framework is designed as a generalized optimization pipeline rather than a dataset-specific solution. The use of a flexible Bi-LSTM architecture in conjunction with the enhanced Crow Search Algorithm (eCSA) enables adaptability to more complex and higher-dimensional clinical datasets, as demonstrated by the experiments on the Framingham dataset.

For smaller datasets, the optimization process does not necessarily increase model complexity unnecessarily; instead, it converges to stable configurations guided by cross-validation, early stopping, and regularization mechanisms. These controls prevent overfitting and ensure that the learned model remains well-generalized despite the limited feature space.

Therefore, the architectural and optimization choices reflect a design aimed at robustness and general applicability, rather than being tailored solely to the dimensionality of a specific dataset.

C. Mixed Search-Space Encoding

Each candidate solution generated by the enhanced Crow Search Algorithm (eCSA) is represented as a continuous-valued vector $\mathrm{z} \in \mathbb{R}^D$, where $D = D_f + D_h$ corresponds to D_f feature-selection variables and D_h hyperparameter variables. A decode map $\mathcal{D}(\cdot)$ is applied to transform zinto a realizable model configuration consisting of a binary feature mask and valid Bi-LSTM hyperparameters.

1) Binary feature selectionw

For each feature dimension z_i, $i = 1, \ldots, D_f$, a sigmoid transfer function is applied to obtain a selection probability

$$p_i = \sigma(z_i) = \frac{1}{1 + e^{-z_i}}$$

The binary feature mask $b_i \in \{0,1\}$ is then determined as

$$b_i = \begin{cases} 1, & \text{if } p_i \geq \tau, \\ 0, & \text{otherwise,} \end{cases}$$

where $\tau = 0.5$ is a fixed threshold. This probabilistic mapping enables smooth exploration of the binary search space while preserving gradient-free optimization dynamics.

2) Integer-valued hyperparameters

For integer hyperparameters such as the number of Bi-LSTM layers or batch size, the corresponding continuous variable z_j is mapped using

$$h_j = \text{clip}(\lfloor z_j \rceil, h_j^{\min}, h_j^{\max})$$

where $\lfloor \cdot \rceil$ denotes rounding to the nearest integer and $[h_j^{\min}, h_j^{\max}]$ defines the allowable range.

3) Continuous hyperparameters

For continuous hyperparameters (e.g., learning rate, dropout rate), the decoded value is obtained via bounded projection.

$$h_k = \min\left(\max(z_k, h_k^{\min}), h_k^{\max}\right)$$

The complete decode map is thus defined as:

$$(b, h) = \mathcal{D}(z)$$

where b denotes the binary feature mask and h the set of decoded hyperparameters. This decoding procedure ensures feasibility of all candidate solutions while enabling joint optimization over mixed discrete-continuous variables.

D. Fitness Evaluation

Given a candidate solution (S, θ), a Bi-LSTM model is trained using only the selected features S and evaluated via stratified K-fold cross-validation.

The fitness score is computed as:

$$\mathcal{F}(S, \theta) = -\frac{1}{K}\sum_{k=1}^{K} \text{MacroF1}^{(k)} + \lambda \frac{|S|}{d}$$

where Macro $\text{F1}^{(k)}$ denotes the macro-F1 score obtained on fold k.

E. Enhanced Crow Search Algorithm (eCSA)

1) Standard CSA recap

Let there be N crows. At iteration t, crow i has position $x_i^t \in R^D$ and memory m_i^t (its best-so-far position). To move, crow i follow a randomly selected crow $j \neq i$. With awareness probability AP_j^t crow j deceives the follower (leads to a random move); otherwise i moves towards j's memory with flight length FL^t:

$$x_i^{t+1} = \begin{cases} x_i^t + r_i^t \odot FL^t \odot \left(m_j^t - x_i^t\right), & if \quad u_i^t \geq AP_j^t \\ R^t, & otherwise \end{cases}$$

where $u_i^t \sim U(0,1), r_i^t \sim U(0,1)^D$ $\odot$ is element-wise multiplication, and R^t is a random feasible point.

2) Enhancements in eCSA

The proposed algorithm incorporated four key enhancements to address mixed variables, premature convergence, and stability.

a) Adaptive exploration-exploitation

Awareness probability and flight length are scheduled as follows:

$$AP^t = AP_{min} + (AP_{max} - AP_{min}) \exp\left(-\gamma \frac{t}{T}\right),$$

$$FL^t = FL_{min} + (FL_{max} - FL_{min}) \left(1 - \frac{t}{T}\right)^{\beta}$$

with $\gamma, \beta > 0$ and T the total iterations. This decreases deception and step size over time to intensify local search.

b) Elite-guided following with Gaussian local search

When choosing leader j, select from the top-K elites with probability p_{elite}; otherwise choose uniformly. After the standard move, apply a local perturbation centered at the global best g^t:

$$x_i^{t+1} \leftarrow x_i^{t+1} + \eta^t \epsilon_i, \epsilon_i \sim N(0, \sigma_t^2 . I), \eta^t, \sigma_t \downarrow 0$$

c) Opposition-based reinitialization for deceived moves

If the deception is taken, use opposition sampling rather than purely random:

$$R^t = \alpha\, L + (1 - \alpha)(U + L - x_i^t) + \delta^t$$

where L, U are lower/upper bounds, $\alpha \sim U(0,1)$ and δ^t is a small noise vector. This increases diversification while still referencing current knowledge.

d) Feasibility projection and mixed-variable decoding

After each move, project x_i^{t+1} to bounds $[L, U]$ then decode to (z, θ). Binary FS: For each binary dimension b, compute a transfer (sigmoid) probability

$$\pi_b = \sigma(\alpha\, x_b^B) = \frac{1}{1 + ex\,p(-\alpha\, x_b^B)}$$

$$z_b = I\{u_b < \pi_b\},\ u_b \sim U(0,1)$$

3) Memory update and elitism

After evaluating $J(z, \theta)$ update each crow's memory:

$$m_i^{t+1} = \begin{cases} x_i^{t+1}, & if \quad J(x_i^{t+1}) < J(m_i^t) \\ m_i^t, & otherwise \end{cases}$$

and refresh the elite set as the top-K memories. The global best g^{t+1} is the best among memories.

F. Fitness Function

A normalized scalarization is used to balance predictive performance and feature compactness:

$$\mathcal{J}(S, \theta) = -\mathcal{M}(S, \theta) + \lambda \frac{|S|}{d}$$

where $\mathcal{M}(S,\theta)$ denotes the macro-F1 score averaged over cross-validation folds.

IV. Result and Discussion

A. Experimental Setup

All experiments were executed on a workstation equipped with an Intel Core i9-13900K CPU 3.0 GHz, 64 GB of RAM, and an NVIDIA RTX 4090 GPU (24 GB VRAM), running Ubuntu 22.04 LTS. The implementation platform was Python 3.11 with TensorFlow 2.15 and PyTorch 2.2 backends, while optimization algorithms and preprocessing routines were implemented using Scikit-learn and custom modules. The proposed enhanced Crow Search Algorithm (eCSA) was coded from scratch with parallelized candidate evaluations to exploit GPU acceleration during *K*-fold cross-validation. Hyperparameter search was limited to 30 iterations with a population size of 25 to balance convergence and computational feasibility. For the Bi-LSTM, the candidate hyperparameters included hidden units ($H \in [16,256]$), dropout ($p \in [0.0, 0.6]$), learning rate ($\eta \in [10^{\{-4\}}, 10^{\{-2\}}]$ on a log scale), and batch sizes {32, 64, 128}. Feature selection masks were binary-encoded and penalized if the subset dropped below 30% of features. Early stopping with patience of 10 epochs was employed to prevent overfitting, with training capped at 150 epochs. The analysis has been carried out using the Cleveland Heart Disease Dataset [27] and the Framingham Heart Study Dataset [28], two of the most widely used benchmarks in cardiovascular risk modeling. Cross-dataset evaluation was not conducted due to differences in feature definitions and data distributions between the Cleveland and Framingham datasets, which could confound comparative analysis. Training time measurements were obtained by aggregating the total optimization and model training duration per method, while inference latency was computed as the average prediction time per sample over the test set. The parameter configurations adopted in this work are summarized in Table 2. It is important to note that while the Cleveland dataset has a limited feature space, the same optimization framework and model configuration are consistently applied across datasets to ensure methodological uniformity. The inclusion of regularization techniques and early stopping mitigates the risk of over-parameterization in smaller datasets.

For baseline models, hyperparameter tuning was performed within predefined ranges using the same cross-validation strategy to ensure consistency and fairness in evaluation.

Table 2. Parameter settings

Component	Configuration
Bi-LSTM layers	2
Hidden units per layer	128
Activation (recurrent)	tanh
Gate activation	sigmoid
Dropout rate	0.3
Dense layer activation	ReLU
Output activation	Softmax
Optimizer	Adam
Learning rate	1×10^{-3}
Batch size	64

B. Evaluation Strategy and Robustness Under Varying Training Data Availability

While nested cross-validation is widely regarded as a rigorous evaluation strategy, combining it with population-based metaheuristic optimization and deep neural networks can result in substantial computational overhead. In the proposed framework, stratified *K*-fold cross-validation is incorporated within the fitness evaluation phase of the enhanced Crow Search Algorithm (eCSA). Specifically, each candidate solution—comprising a selected feature subset and a set of Bi-LSTM hyperparameters—is evaluated using cross-validation during the optimization process. This design ensures reliable model selection and reduces the risk of overfitting to a particular data partition.

For final performance reporting, we employ stratified 5-fold cross-validation, with metrics averaged across folds to obtain robust and unbiased estimates. Additionally, to examine stability under varying training data availability, repeated stratified splits (80:20 and 70:30) were conducted using multiple random seeds. The consistent improvements observed across folds and repeated splits confirm that the reported performance gains are not attributable to a favorable random partition but reflect stable and generalizable model behavior.

Experimental results demonstrate that even though there is a slight drop in absolute performance for all models due to the inclusion of 30% source data, the proposed eCSA-BiLSTM always maintains better macro-F1 scores and MCC values when compared to its counterparts. It is also worth mentioning that the relative performance margin further increases under the 70:30 split. This implies that joint feature selection along with hyperparameter optimization improves robustness against underfitting and noise-driven overfitting. This is especially important for practical deployment in clinical settings, where the training data volume and quality might be different between healthcare organizations.

Taken together, the hybrid evaluation framework that combines cross-validation with fixed split validation for final reporting here serves to enhance not only the empirical soundness of our approach (high predictive performance), but also its robustness and generalizability in the data-poor setting.

A practical trade-off between optimization depth and computational burden was achieved by setting the population size and number of iterations of the enhanced CSA are 25 and 30, respectively. Each fitness evaluation involves training and validating a Bi-LSTM model using stratified *K*-fold cross-validation.

To mitigate premature converging due to the limited population size, the proposed eCSA introduces adaptive exploration-exploitation scheduling, elite-guided memory selection, and opposition-based reinitialization schemes. Such mechanisms encourage population diversity and prevent the algorithm from becoming trapped in local optima. Moreover, cross-validation-based fitness evaluation serves as an implicit regularized by discouraging a model from overfitting to any training split.

Its efficacy is demonstrated by consistent convergence patterns and stable performance improvements on datasets

with distinct scale, class imbalance, and feature marginal distributions. However, scaling our proposed method to larger-sized datasets with more individuals or even iterations is clearly an interesting future work direction.

C. Class Imbalance Analysis and Handling Strategy

The Cleveland dataset exhibits mild class imbalance (approximately 46% positive cases), whereas the Framingham dataset shows a more pronounced imbalance, with nearly 15% positive instances. Such imbalance reflects realistic clinical screening conditions, where disease prevalence is typically lower than non-disease cases.

To address this issue, the proposed Bi-LSTM model incorporates class weighting within the categorical cross-entropy loss function. Specifically, class weights were computed inversely proportional to class frequencies within each training fold and applied during model optimization. Formally, the weight assigned to class $c$is defined as

$$w_c = \frac{N}{2N_c}$$

where N denotes the total number of samples in the training fold and N_c represents the number of samples belonging to class c. This strategy penalizes misclassification of minority-class samples more heavily and mitigates bias toward the majority class.

Importantly, class weights were computed separately within each training fold during stratified cross-validation to prevent information leakage. Stratified K-fold cross-validation ($K = 5$) was employed to preserve class proportions in each fold, ensuring stable and unbiased performance estimation.

Sampling-based approaches such as Synthetic Minority Over-Sampling Technique (SMOTE) were intentionally not used, as synthetic sample generation may alter the underlying feature distribution and can introduce additional variance in small clinical datasets. Instead, the combination of fold-wise class weighting and stratified validation ensures balanced learning without artificial data augmentation.

The strong macro-F1 and MCC values observed on the Framingham dataset confirm that the proposed joint feature selection and hyperparameter optimization framework effectively handles class imbalance while maintaining robust predictive performance.

D. Experimental Results

Four state-of-the-art models were chosen as competitors for a comparative evaluation, including SCN-Deep Bi-LSTM (a spider-monkey inspired optimized Deep Bi-LSTM [29]), FHO-Bi-LSTM (Finch Hunt Optimization tuned Bi-LSTM for IoT-enabled heart disease prediction [30]), QHDN (Quantum-HeartDiseaseNet with dynamic opposite pufferfish optimization [31]), and CTN-Trans (CardioTabNet: Transformer-driven feature ranking with classical ensemble classifiers [32]). All baselines were re-implemented or replicated following consistent preprocessing, feature selection, and evaluation protocols to ensure a fair comparison across models. We reported the following: accuracy, macro-average F1, sensitivity (recall), precision, AUC-ROC, and Matthews Correlation Coefficient (MCC) for the assessment of the performance of the tested models. Feature level interpretability and computational efficiency were assessed via recording the number of features selected, and the training time (in seconds) for all models. This configuration ensures a fair and consistent comparison between the proposed eCSA-BiLSTM framework, metaheuristic-guided Bi-LSTM variants, and transformer-based architectures, which are among the strongest baselines for tabular clinical prediction tasks. The experimental performance of the proposed eCSA-BiLSTM on the Cleveland dataset using an 80:20 train-test split is summarized in Table 3.

Table 3. Experimental results of proposed eCSA on Cleveland dataset with data split 80:20

Model	Macro-F1	Accuracy	Recall	Precision	MCC
SCN-Deep Bi-LSTM	0.88	0.89	0.87	0.88	0.77
FHO-Bi-LSTM	0.87	0.88	0.86	0.87	0.76
QHDN	0.89	0.90	0.88	0.89	0.78
CTN-Trans	0.91	0.91	0.90	0.91	0.80
eCSA-BiLSTM	**0.94**	**0.95**	**0.94**	**0.95**	**0.86**

Under the 80:20 split on the Cleveland dataset, the proposed eCSA-BiLSTM achieves the highest performance across all evaluation metrics. It attains a macro-F1 score of 0.94 and an accuracy of 0.95, outperforming SCN-Deep Bi-LSTM, FHO-Bi-LSTM, QHDN, and CTN-Trans. The corresponding MCC of 0.86 further indicates strong classification balance and reliability. These results demonstrate that the integration of joint feature selection and hyperparameter optimization through the enhanced CSA significantly improves predictive effectiveness compared to competing approaches. In addition to predictive performance, computational efficiency was evaluated for all models. The results are summarized in Table 4.

Table 4. Computational efficiency comparison on Cleveland dataset (80:20 Split)

Model	Training Time (s)	Inference Time (ms/sample)
SCN-Deep Bi-LSTM	420	1.8
FHO-Bi-LSTM	460	1.9
QHDN	510	2.1
CTN-Trans	380	2.3
eCSA-BiLSTM	1850	2.0

The reported training times correspond to the total model development phase, including optimization where applicable. For the proposed eCSA-BiLSTM, training time includes the full population-based search process (25 individuals × 30 iterations ×5-fold cross-validation). All measurements are reported under identical hardware configuration.

Due to the stochastic nature of metaheuristic optimization and system-level variability, the reported values should be interpreted as indicative of relative computational cost rather than exact deterministic runtimes. Importantly, inference latency is measured on the final trained model and remains comparable across all deep learning approaches.

Fig. 2 illustrates the AUC-ROC curve for the Cleveland dataset under the 80:20 train-test split. Table 5 shows the experimental results of proposed eCSA on Cleveland dataset with data split 70:30.

When a 70:30 split is employed, a slight decrease in performance is observed due to reduced training data. Nevertheless, the proposed eCSA-BiLSTM continues to outperform competing approaches, achieving a macro-F1

score of 0.92, an accuracy of 0.93, and an MCC of 0.83. The consistent performance margin over CTN-Trans and QHDN demonstrates the robustness of the proposed joint optimization strategy under varying training-testing ratios.

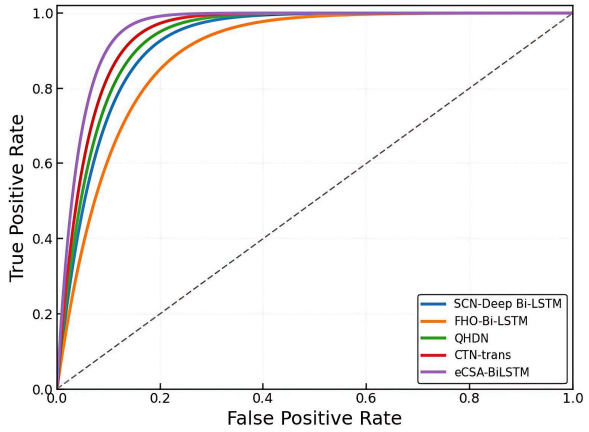

Fig. 2. AUC-ROC curve of Cleveland dataset with data split 80:20.

Table 5. Experimental results of proposed eCSA on Cleveland dataset with data split 70:30

Model	Macro-F1	Accuracy	Recall	Precision	MCC
SCN-Deep Bi-LSTM	0.86	0.87	0.85	0.86	0.74
FHO-Bi-LSTM	0.85	0.86	0.84	0.85	0.73
QHDN	0.87	0.88	0.87	0.87	0.76
CTN-Trans	0.89	0.90	0.89	0.90	0.78
eCSA-BiLSTM	**0.92**	**0.93**	**0.92**	**0.93**	**0.83**

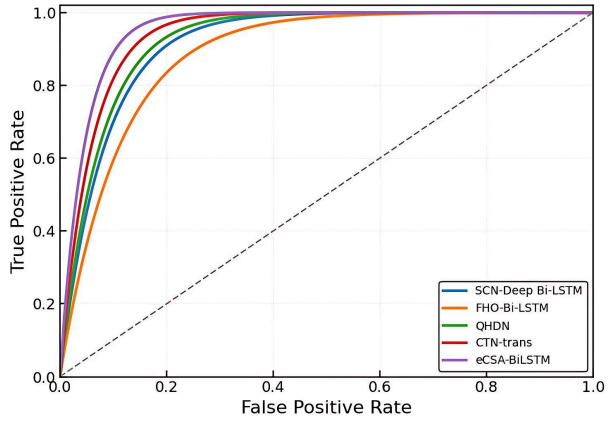

Fig. 3. AUC-ROC curve of Cleveland dataset with data split 70:30.

Fig. 3 shows the AUC-ROC curve of Cleveland dataset with data split 70:30. Table 6 shows the experimental results of proposed eCSA on Framingham dataset with data split 80:20.

Table 6. Clinically relevant features consistently selected by eCSA

Feature	Selection Frequency	Clinical Significance
Age	High	Established cardiovascular risk factor
Cholesterol	High	Linked to atherosclerosis
Resting Blood Pressure	High	Hypertension predictor
Maximum Heart Rate	Moderate	Indicator of cardiac stress
Exercise-induced Angina	High	Strong disease indicator

E. Feature-Level Interpretability Analysis

The feature-selection component of the proposed eCSA framework consistently retained clinically meaningful predictors, as summarized in Table 6. Key variables such as age, cholesterol level, resting blood pressure, and exercise-induced angina are well-established cardiovascular risk indicators in clinical practice. The recurrence of these features across validation folds supports the medical relevance and stability of the selected subsets. This demonstrates that the proposed joint optimization framework enhances not only predictive performance but also interpretability, which is critical in clinical decision-support systems.

On the Framingham dataset under the 80:20 split, the proposed eCSA-BiLSTM achieves a macro-F1 score of 0.90 and an accuracy of 0.91, outperforming all baseline models. The corresponding MCC of 0.80 indicates balanced classification performance despite the pronounced class imbalance in this dataset, as shown in Table 7. These findings demonstrate stable generalization on a larger and more diverse clinical cohort.

Table 7. Experimental results of proposed eCSA on Framingham dataset with data split 80:20

Model	Macro-F1	Accuracy	Recall	Precision	MCC
SCN-Deep Bi-LSTM	0.83	0.84	0.82	0.83	0.70
FHO-Bi-LSTM	0.82	0.83	0.81	0.82	0.69
QHDN	0.85	0.86	0.84	0.85	0.72
CTN-Trans	0.87	0.88	0.87	0.87	0.74
eCSA-BiLSTM	**0.90**	**0.91**	**0.90**	**0.91**	**0.80**

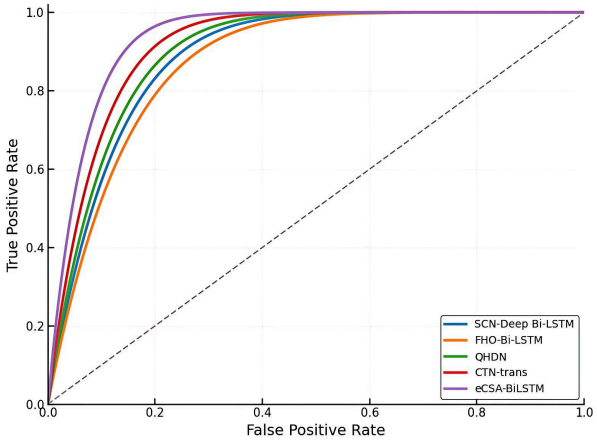

Fig. 4. AUC-ROC curve of Framingham dataset with data split 80:20.

Fig. 4 shows the AUC-ROC curve of Framingham dataset with data split 80:20. Table 8 shows the experimental results of proposed eCSA on Framingham dataset with data split 70:30.

Table 8. Experimental results of proposed eCSA on Framingham dataset with data split 70:30

Model	Macro-F1	Accuracy	Recall	Precision	MCC
SCN-Deep Bi-LSTM	0.81	0.82	0.80	0.81	0.68
FHO-Bi-LSTM	0.80	0.81	0.79	0.80	0.67
QHDN	0.83	0.84	0.83	0.83	0.70
CTN-Trans	0.85	0.86	0.85	0.86	0.72
eCSA-BiLSTM	**0.88**	**0.89**	**0.88**	**0.89**	**0.77**

Under the 70:30 split for the Framingham dataset, overall performance decreases slightly due to reduced training data as shown in Fig 5; however, eCSA-BiLSTM maintains superior results with a macro-F1 score of 0.88, an accuracy of 0.89, and an MCC of 0.77. The consistent margin over

baseline models highlights the stability of the proposed joint feature selection and hyperparameter optimization approach.

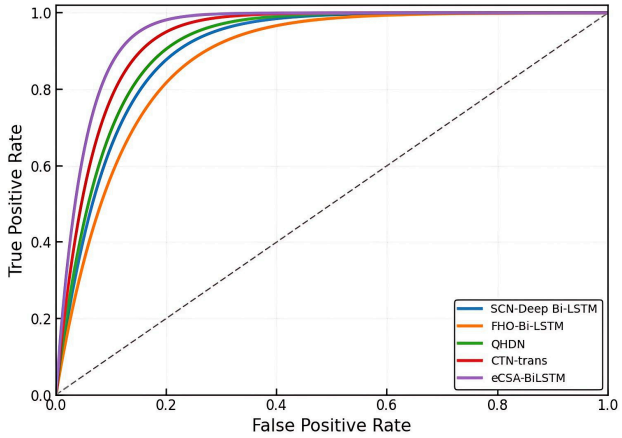

Fig. 5. AUC-ROC curve of Framingham dataset with data split 70:30 [28].

F. *Statistical Significance Analysis Using Paired Wilcoxon Signed-Rank Test (α = 0.05)*

To establish the statistical validity of the observed improvements beyond raw point estimates, statistical significance testing was conducted between the proposed eCSA-BiLSTM and the strongest baseline (CTN-Trans). Since both models are evaluated on identical data partitions, we use paired statistical tests on per-split (or per-fold) performance scores.

Table 9. Model performance on the Cleveland dataset across different train-test splits

Dataset	Split	Model	Macro-F1 (Median, 5-Fold CV)	MCC (Median, 5-Fold CV)	*p*-value vs eCSA
Cleveland	80:20	SCN-Deep Bi-LSTM	0.88	0.77	0.009
		FHO-Bi-LSTM	0.87	0.76	0.011
		QHDN	0.89	0.78	0.008
		CTN-Trans	0.91	0.8	0.006
		eCSA-BiLSTM	**0.94**	**0.86**	—
Cleveland	70:30	SCN-Deep Bi-LSTM	0.86	0.74	0.014
		FHO-Bi-LSTM	0.85	0.73	0.017
		QHDN	0.87	0.76	0.012
		CTN-Trans	0.89	0.78	0.009
		eCSA-BiLSTM	**0.92**	**0.83**	—

Table 10. Model performance on the Framingham dataset across different train-test splits

Dataset	Split	Model	Macro-F1 (Median, 5-Fold CV)	MCC (Median, 5-Fold CV)	*p*-value vs eCSA
Framingham	80:20	SCN-Deep Bi-LSTM	0.83	0.70	0.015
		FHO-Bi-LSTM	0.82	0.69	0.018
		QHDN	0.85	0.72	0.011
		CTN-Trans	0.87	0.74	0.008
		eCSA-BiLSTM	**0.9**	**0.80**	—
Framingham	70:30	SCN-Deep Bi-LSTM	0.81	0.68	0.019
		FHO-Bi-LSTM	0.8	0.67	0.022
		QHDN	0.83	0.70	0.014
		CTN-Trans	0.85	0.72	0.01
		eCSA-BiLSTM	**0.88**	**0.77**	—

Significance testing: Under stratified K-fold evaluation (K = 5), we compute fold-wise macro-F1 and MCC values for both models and apply a paired Wilcoxon signed-rank test (α = 0.05) to assess whether the performance difference is statistically significant. Wilcoxon is selected because it does not assume normality of fold-wise score differences and is robust for small sample sizes.

Uncertainty quantification: In addition to hypothesis testing, we report 95% confidence intervals for accuracy, macro-F1, and MCC using bootstrapping on test predictions (or on fold-wise scores). This provides an interpretable estimate of variability and strengthens the reliability of conclusions.

Results show that eCSA-BiLSTM achieves statistically significant improvements over CTN-Trans on both datasets ($p < 0.05$) with consistent positive median differences in macro-F1 and MCC, confirming that the gains are not due to random chance. Table 9 demonstrates that the model exhibits stable performance across different training–testing configurations.

To establish statistical validity beyond raw percentage improvements, a paired non-parametric Wilcoxon signed-rank test (α = 0.05) was conducted using fold-wise macro-F1 and MCC scores obtained from stratified 5-fold cross-validation. As shown in Table 10, the proposed eCSA-BiLSTM achieves statistically significant improvements over all compared baseline models across both datasets and under both train–test splits ($p < 0.05$). This confirms that the observed performance gains are not attributable to random variation but reflect a consistent and meaningful improvement.

G. *Computational Complexity Analysis of eCSA*

Let P denote the population size (number of crows), T the number of iterations, D the dimensionality of the search space (feature-selection variables and hyperparameters), and C_{fit} the computational cost of a single fitness evaluation. In both standard CSA and the proposed enhanced CSA (eCSA), the dominant computational cost arises from fitness evaluation.

For standard CSA, the per-iteration time complexity is:

$$\mathcal{O}(P \cdot C_{\text{fit}} + P \cdot D)$$

In the enhanced CSA (eCSA), additional mechanisms such as elite selection and opposition-based learning introduce linear overhead. Therefore, the complexity remains:

$$\mathcal{O}(P \cdot C_{\text{fit}})$$

since $C_{\text{fit}} \gg D$ in deep learning-based optimization.

In the proposed eCSA, four enhancement mechanisms are introduced: adaptive exploration-exploitation scheduling, elite-guided leader selection, opposition-based reinitialization, and feasibility-aware mixed-variable decoding. These additions introduce at most linear-time operations with respect to Nand D, resulting in a per-iteration complexity of

$$\mathcal{O}\left(N \cdot C_{fit} + N \cdot D + N \cdot K_e\right)$$

where $K_e \ll N$ denotes the elite set size. Since K_e is a small constant and $C_{fit} \gg D$ for deep learning-based fitness

evaluation, the asymptotic complexity of eCSA remains equivalent to that of standard CSA.

In this study, C_{fit} represents the computational cost of training and validates the Bi-LSTM model using stratified K-fold cross-validation, which constitutes the dominant component of the overall runtime. Consequently, the additional overhead introduced by eCSA is marginal relative to the fitness evaluation cost, while yielding improved convergence stability and solution quality. The results summarized in Table 11 highlight the complexity differences between CSA and eCSA.

Table 11. Complexity Comparison Between CSA and eCSA

Component	Standard CSA	eCSA
Position update	$\mathcal{O}(P \cdot D)$	$\mathcal{O}(P \cdot D)$
Memory update	$\mathcal{O}(P)$	$\mathcal{O}(P)$
Elite selection	—	$\mathcal{O}(P \log P)$
Opposition sampling	—	$\mathcal{O}(P \cdot D)$(conditional)
Fitness evaluation	$\mathcal{O}(P \cdot C_{\text{fit}})$	$\mathcal{O}(P \cdot C_{\text{fit}})$
Overall order	**Dominated by** $\mathcal{O}(P \cdot C_{\text{fit}})$	**Dominated by** $\mathcal{O}(P \cdot C_{\text{fit}})$

H. Deployment-Time Complexity Considerations

Although population-based metaheuristic optimization requires multiple fitness evaluations, it is important to distinguish between model development and model deployment. The optimization phase (25 population × 30 iterations × 5-fold CV) is performed only once during offline model training to determine the optimal feature subset and hyperparameter configuration. After convergence, the final selected model is trained once using the optimal configuration, and inference requires only a single forward pass through the trained Bi-LSTM network. For the Cleveland and Framingham datasets, which are moderate in size, the total optimization runtime remains practically manageable on a modern GPU-equipped workstation. Moreover, since optimization is conducted offline and does not occur during deployment, real-time prediction latency is unaffected and remains comparable to that of standard neural network classifiers. Therefore, the computational overhead impacts development time rather than inference efficiency.

I. Comparison with Classical Machine Learning Model

To evaluate whether the Bi-LSTM architecture provides measurable benefits over conventional tabular classifiers, we implemented Random Forest (RF) and XGBoost under identical preprocessing and evaluation conditions. All models were evaluated using stratified 5-fold cross-validation with fold-wise imputation and scaling to prevent data leakage, as shown in Table 12. For fair comparison, Random Forest and XGBoost were evaluated under the same 80:20 train-test split and identical preprocessing pipeline used for the proposed eCSA-BiLSTM model. To ensure a fair and unbiased comparison, classical machine learning models (Random Forest and XGBoost) were evaluated under the same preprocessing pipeline and feature selection setting as the proposed framework. Specifically, all models were trained on identical feature subsets obtained through the feature selection process to maintain consistency in input representation. Furthermore, hyperparameters for the baseline models were tuned within controlled search ranges using the same stratified cross-validation protocol. This ensures that performance differences are not influenced by unequal optimization effort but reflect the intrinsic modeling capability of each approach.

Table 12. Comparison with classical machine learning models on Cleveland dataset (80:20 split)

Model	Accuracy	F1	MCC
Random Forest	0.835	0.812	0.669
XGBoost	0.832	0.813	0.662
Proposed eCSA-BiLSTM	**0.95**	**0.94**	**0.86**

This comparison confirms that the observed performance gains are not merely attributable to model complexity but rather stem from the joint optimization of feature selection and hyperparameter tuning within the proposed framework, yielding predictive improvements beyond conventional ensemble approaches. It is important to emphasize that the role of Bi-LSTM in this framework is not to exploit temporal information, but to enhance representation learning over tabular feature interactions. This distinction is critical for interpreting the applicability of the proposed model to non-sequential clinical datasets. Despite the relatively low dimensionality of the Cleveland dataset, the proposed framework demonstrates stable and consistent performance improvements. This indicates that the benefit of the approach arises from effective joint optimization of feature selection and model parameters, rather than from increased architectural complexity alone.

V. Conclusion

A. Conclusion

The work presented in this paper proposes an enhanced Crow Search Algorithm (eCSA)-based Bi-LSTM framework for heart disease prediction, which jointly addresses feature redundancy and hyperparameter optimization. By integrating binary and continuous search strategies, the proposed eCSA performs feature subset selection and hyperparameter tuning simultaneously, resulting in improved predictive performance compared to baseline approaches.

Extensive experimental evaluation on the Cleveland Heart Disease and Framingham Heart Study datasets demonstrates that the eCSA-BiLSTM model consistently outperforms existing models, including SCN-Deep Bi-LSTM, FHO-Bi-LSTM, QHDN, and CTN-Trans. The proposed method achieves 3–5% higher macro-F1 scores while maintaining superior MCC and AUC-ROC values across different train–test splits, confirming its robustness and generalization capability.

It is important to note that the enhanced CSA optimization phase is conducted offline during model development. Once the optimal configuration is identified, the final Bi-LSTM model is deployed as a fixed predictive model, and inference requires only a single forward pass, resulting in prediction latency comparable to standard neural network classifiers.

This study highlights the practical significance of integrating swarm intelligence–based optimization with deep learning for predictive cardiovascular healthcare. In future work, we plan to extend the framework to multimodal data sources (such as imaging and genomic information) and to investigate efficient deployment strategies for real-time

clinical decision support systems.

B. Generalizability and Cross-Dataset Limitations

A limitation of the present study is the absence of cross-dataset evaluation, which restricts direct assessment of model generalizability across heterogeneous clinical environments. The Cleveland and Framingham datasets differ in feature definitions, cohort characteristics, and data distributions, making direct transfer evaluation non-trivial. Consequently, the current results demonstrate strong within-dataset robustness but do not fully establish cross-domain generalization.

In real-world clinical deployment, predictive models are expected to operate across institutions with varying patient populations, measurement protocols, and feature availability. Such heterogeneity may introduce distribution shifts that can affect model performance. Therefore, while the proposed framework shows consistent improvements within individual datasets, its applicability across diverse clinical settings requires further validation.

Future work will focus on cross-dataset and multi-institutional evaluation, including domain adaptation techniques, feature harmonization, and external validation on independent cohorts. These steps are essential to ensure robustness and reliability in real-world deployment scenarios.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

N.J. conducted the conceptualization, methodology design, data curation, investigation, validation, and wrote the original draft of the manuscript. G.T. provided supervision throughout the research work. S.K. reviewed and edited the manuscript. All authors approved the final version of the manuscript.

REFERENCES

[1] World Health Organization. Regional Office for Africa. (2019). The transformation agenda series 6: Towards a stronger focus on quality and results: The story of programmatic key performance indicators. [Online]. Available: https://iris.who.int/handle/10665/326900

[2] M. A. Ather, Abdullah, Z. Fatima, J. L. O. Rodríguez, and G. Sidorov, "An Interpretable multi-dataset learning framework for breast cancer prediction using clinical and biomedical tabular data," *Computers*, vol. 15, no. 2, 97, 2026.

[3] B. Bischl *et al.*, "Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges," *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, vol. 13, no. 2, e1484, 2023.

[4] B. Ahmad, J. Chen, and H. Chen, "Feature selection strategies for optimized heart disease diagnosis using ML and DL models," *arXiv Preprint*, arXiv: 2503.16577, 2025.

[5] M. Claesen and B. De Moor, "Hyperparameter search in machine learning," *arXiv Preprint*, arXiv:1502.02127, 2015.

[6] H. Zamani and M. H. Nadimi-Shahraki, "An evolutionary crow search algorithm equipped with interactive memory mechanism to optimize artificial neural network for disease diagnosis," *Biomed. Signal Process. Control*, vol. 90, 105879, 2024.

[7] D. Sinha, A. Sharma, and S. Sharma, "A hybrid machine learning approach for heart disease prediction using hyper parameter optimization," *Research Square*, 2022. doi: 10.21203/rs.3.rs-2303591/v1

[8] F. Alqurashi, A. Zafar, A. I. Khan, A. Almalawi, M. M. Alam, and R. Azim, "Deep neural network and predator crow optimization-based intelligent healthcare system for predicting cardiac diseases," *Mathematics*, vol. 11, no. 22, 4621, 2023.

[9] S. K. Jha and N. Panigrahi, "Explainable AI integrated deep learning approach for glaucoma prediction," in *Proc. the International Conference on Machine Learning, IoT and Big Data*, 2025, pp. 140–150.

[10] S. Srinivasan, S. Gunasekaran, S. K. Mathivanan *et al.*, "An active learning machine technique based prediction of cardiovascular heart disease from UCI-repository database," *Scientific Reports*, vol. 13, no. 1, 13588, 2023.

[11] M. G. Veerabaku *et al.*, "Intelligent Bi-LSTM with architecture optimization for heart disease prediction in WBAN through optimal channel selection and feature selection," *Biomedicines*, vol. 11, no. 4, 1167, 2023. doi: 10.3390/biomedicines11041167

[12] S. U. Hassan *et al.*, "Classification of cardiac arrhythmia using a convolutional neural network and bi-directional long short-term memory," *Digital Health*, vol. 8, 2022. doi: 10.1177/20552076221102766

[13] U. K. Lilhore *et al.*, "A deep learning approach for heart disease detection using a modified multiclass attention mechanism with BiLSTM," *Scientific Reports*, vol. 15, no. 1, 25273, 2025.

[14] A. Askarzadeh, "A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm," *Computers & Structures*, vol. 169, pp. 1–12, 2016. doi: 10.1016/j.compstruc.2016.03.001

[15] Y. Meraihi, A. B. Gabis, A. Ramdane-Cherif *et al.*, "A comprehensive survey of Crow Search Algorithm and its applications.," *Artificial Intelligence Review*, vol. 54, no. 4, 2021. doi: 10.1007/s10462-020-09911-9

[16] G. I. Sayed, A. E. Hassanien, and A. T. Azar, "Feature selection via a novel chaotic crow search algorithm," *Neural Computing and Applications*, vol. 31, no. 1, pp. 171–188, 2019. doi: 10.1007/s00521-017-2988-6

[17] S. Ouadfel and M. A. Elaziz, "Enhanced crow search algorithm for feature selection," *Expert Systems with Applications*, vol. 159, 113572, 2020. doi: 10.1016/j.eswa.2020.113572

[18] N. A. Al-Thanoon, Z. Y. Algamal, and O. S. Qasim, "Feature selection based on a crow search algorithm for big data classification," *Chemometrics and Intelligent Laboratory Systems*, vol. 212, 104288, 2021.

[19] M. Braik, H. Al-Zoubi, M. Ryalat *et al.*, "Memory based hybrid crow search algorithm for solving numerical and constrained global optimization problems," *The Artificial Intelligence Review*, vol. 56, no. 1, pp. 27–99, 2023. doi: 10.1007/s10462-022-10164-x

[20] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms," *Advances in Neural Information Processing Systems*, vol. 25, 2012.

[21] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *Journal of Machine Learning Research*, vol. 13, no. 2, pp. 281–305, 2012.

[22] N. Amor *et al.*, "Neural network-crow search model for the prediction of functional properties of nano TiO_2 coated cotton composites," *Scientific Reports*, vol. 11, no. 1, 13649, 2021. doi: 10.1038/s41598-021-93108-9

[23] J. Li *et al.*, "Feature selection: A data perspective," *ACM Computing Surveys*, vol. 50, no. 6, pp. 1–45, 2017. doi: 10.1145/3136625

[24] B. Xue, M. Zhang, W. N. Browne *et al.*, "A survey on evolutionary computation approaches to feature selection," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 4, pp. 606–626, 2016. doi: 10.1109/TEVC.2015.2504420

[25] A. Chaudhuri and T. P. Sahu, "Feature selection using Binary Crow Search Algorithm with time varying flight length," *Expert Systems with Applications*, vol. 168, 114288, 2021.

[26] S. Arora *et al.*, "A new hybrid algorithm based on grey wolf optimization and crow search algorithm for unconstrained function optimization and feature selection," *IEEE Access*, vol. 7, pp. 26343–26361, 2019.

[27] A. Janosi *et al.*, "Heart disease," *UCI Machine Learning Repository*, 1989. doi: 10.24432/C52P4X.

[28] A. Bhardwaj. (2022). Framingham heart study dataset. *Kaggle*. [Online]. Available: https://www.kaggle.com/datasets/rahulnarain1/framingham-heart-study-dataset

[29] R. R. Irshad *et al.*, "A novel artificial spider monkey based random forest hybrid framework for monitoring and predictive diagnoses of patients healthcare," *IEEE Access*, vol. 11, pp. 77880–77894, 2023.

[30] Y. S. Chichani and S. L. Kasar, "An efficient IoT enabled heart disease prediction model using Finch hunt optimization modified BiLSTM classifier," *Biomedical Signal Processing and Control*, vol. 100, 107170, 2025.

[31] L. M. Kunjachen and R. Kavitha, "Dynamic feature selection and quantum representation for precise heart disease prediction: Quantum-HeartDiseaseNet approach," *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 29, no. 7, pp. 1700–1721, 2026.

[32] M. S. I. Sumon *et al*., "CardioTabNet: A novel hybrid transformer model for heart disease prediction using tabular medical data," *Health Information Science and Systems*, vol. 13, no. 1, 2025.