Frontiers in Health Informatics
ISSN-Online: 2676-7104
2024; Vol 13: Issue 2

www.healthinformaticsjournal.com
Open Access

Refining Deep Learning Precision in Cardiovascular Risk Assessment: Leveraging Meta-Heuristic Optimization Techniques for Hyperparameter and Architecture Fine-Tuning

[1]Narawade Vaibhav Eknath, [2]Krishna Prasad K,

[1]Post-Doctoral Research Fellow, Institute of Computer Science and Information Science, Srinivas University, Pandeshwar, Mangaluru-575001, Karnataka, India. vnarawade@gmail.com

[2]Professor and Head, Department of Cyber Security and Cyber Forensics, Institute of Engineering and Technology, Srinivas University, Mukka, Mangaluru-574146, Karanataka, India. krishnaprasadkcci@srinivasuniversity.edu.in

Article Info

Article type:

Research

Article History:

Received: 2024-04-04

Revised: 2024-05-29

Accepted: 2024-06-20

Keywords:

Cardiovascular Risk Assessment, Deep Learning Optimization, Meta-Heuristic Techniques, Hyperparameter Tuning, Genetic Algorithms, Predictive Accuracy in Healthcare

ABSTRACT

Cardiovascular diseases (CVDs) are still the top cause of death in the world, so we need more improved ways to figure out who is at risk early on. Deep learning (DL) has appeared potential in this range, but its exactness depends on how well the hyperparameters and organize plan are tuned. This consider looks at how to move forward DL models for figuring out cardiovascular hazard by utilizing meta-heuristic optimization strategies. These strategies, which are based on common and developmental forms, give dependable ways to discover your way through the tremendous and complicated look spaces that come with setting hyperparameters and design. To induce the most excellent comes about from the DL show settings, we utilized a blend of hereditary calculations (GAs), molecule swarm optimization (PSO), and reenacted strengthening (SA). We needed to move forward the model's capacity to foresee and apply to other circumstances by utilizing these meta-heuristic strategies. This would offer assistance us get around the issues with standard framework and irregular look strategies. The test utilized in this think about had a part of diverse clinical and statistic components, which made beyond any doubt that the hazard appraisal show was total. The results of our tests appeared that the models worked much way better, with the most excellent profound learning plans being able to recognize cardiovascular hazard with higher exactness, affectability, and exactness. The meta-heuristic strategies were able to discover the leading hyperparameters, which decreased overfitting and made the show more solid over distinctive populace bunches. The comparison too appeared that combining these methods worked way better than utilizing fair one optimization strategy. This appears how valuable multi-strategy approaches can be in DL advancement. This think about appears how meta-heuristic optimization can totally alter the precision of DL in figuring out cardiovascular hazard. We got way better anticipated execution by fine-tuning hyperparameters and plan. This made CVD risk classification more accurate and useful. More research needs to be done on how to use the model in real-life clinical settings and on finding more meta-heuristic methods to make the model even more accurate and stable.

1. INTRODUCTION

Heart diseases and strokes kill millions of people every year, making them the top cause of death in the world. To stop these diseases from happening and lessen their effects, it is important to do an early and accurate risk assessment. Traditional ways of figuring out a person's cardiovascular risk depend on clinical knowledge and standard statistical models. These models work, but they're not always accurate enough for personalized patient care. Profound learning (DL) has ended up a really valuable instrument in therapeutic determination in later a long time, permitting for more precise analyze and way better forecasts than ever some time recently. But the victory of DL models depends a parcel on how well the hyperparameters are tuned and how the arrange topologies are outlined [1]. Since this prepare is complicated and employments a parcel of computing control, it needs advanced tuning strategies to induce the leading comes about. Meta-heuristic optimization strategies, which are based on characteristic and developmental forms, offer a solid way to bargain with the troubles of fine-tuning DL models' hyperparameters and plan. A few of these strategies, like hereditary calculations (GAs), molecule swarm optimization (PSO), and reenacted tempering (SA), are great for investigating the tremendous look spaces that come with DL optimization. Propelled by the tempering handle in metals, simulated annealing uses a random method to look the space, permitting for incidental moves tough to induce out of nearby optima. Each of these meta-heuristic strategies has its claim benefits, and combining them can make optimization indeed superior [2].

We need to move forward DL models for figuring out cardiovascular chance in this consider by utilizing these meta-heuristic optimization strategies. We need to induce superior precision and generalizability in our forecasts by carefully setting the hyperparameters and structures of our DL models. The dataset we utilized for our thinks about features a part of diverse clinical and statistic variables, such as the age and sex of the patients, their blood weight and cholesterol levels, whether they smoke, and whether they have a family history of CVDs. This huge set of information makes it conceivable to form a chance evaluation apparatus that takes under consideration numerous variables that influence heart wellbeing [3]. Our method employments a refining handle with a few steps. We begin by utilizing each meta-heuristic strategy on its own to discover the finest hyperparameter settings and network designs. Then, we utilize cross-validation to test the execution of these combinations to form beyond any doubt they are steady and maintain a strategic distance from overfitting. After that, we look into mixed optimization methods that utilize parts of GAs, PSOs, and SA to make our models indeed way better. With this strategy, the finest parts of each method are utilized together to create the look region more productively explored and way better answers found [4].

Meta-heuristic optimization makes DL models much way better at figuring out cardiovascular hazard, concurring to the comes about of our tests. Superior accuracy, sensitivity, and specificity are seen in optimized models, which make it conceivable to form more precise expectations of cardiovascular chance [22]. Our improved DL models make it possible for more effective and personalized patient care by making predictions that are more accurate and applicable to more situations. In the future, more research will be done on how to use these models in real-life clinical settings and on finding more meta-heuristic methods to make them even more reliable and effective. A big step forward in the fight against cardiovascular diseases is this study, which uses deep learning and meta-heuristic optimization to their fullest.

2. RELATED WORK

In later a long time, there has been a part of intrigued in utilizing profound learning (DL) to figure out the chance of heart illness. A parcel of inquire about has appeared that DL models may be way better at distinguishing cardiovascular maladies (CVDs) than standard measurable strategies. For case, convolutional neural systems (CNNs) and repetitive neural systems (RNNs) have been utilized by analysts to see at clinical information and imaging strategies and make exceptionally precise hazard forecasts [5, 6]. Indeed with these changes, the exactness of deep learning models still depends on picking the finest hyperparameters and arrange topologies, which is still difficult to do since the look space is so enormous and complicated. It has gotten to be clear that meta-heuristic optimization strategies are exceptionally valuable for fathoming the issues that come up when attempting to tune hyperparameters and move forward the plan of DL models [7]. This strategy has been utilized effectively to progress DL models for a number of restorative purposes, such as figuring out the chance of CVD [8, 9]. On the off chance that you see at how feathered creatures run and angle school, you'll be able get thoughts for another valuable meta-heuristic method utilized in DL optimization. PSO calculations move particles around within the look space based on their possess encounters and the experiences of their companions. This makes a difference them discover the most excellent answers [10]. Thinks about have appeared that PSO can effectively make strides the execution of hyperparameters in DL models, making them way better at making restorative analyze [11, 12].

A random strategy is utilized to investigate the look region in recreated toughening (SA), which is just like the tempering prepare in metalworking. SA can get out of neighborhood optima and move toward worldwide optima [13] by letting it make a few tough moves. The victory of models in a number of therapeutic employments has been incredibly progressed utilizing this way to change DL plans and hyperparameters [14, 15]. To progress optimization performance indeed more, the blending of meta-heuristic strategies has too been looked into. By combining parts of GAs, PSOs, and SAs, the finest highlights of each can be utilized, making the look space investigation quicker and superior answers found. For occasion, blended optimization methods have been utilized on DL models to distinguish infections, and the comes about are superior than with person optimization strategies [16, 17]. Several studies have looked at how to combine meta-heuristic optimization strategies with DL models to figure out the risk of heart illness. Analysts have utilized GAs to make strides the precision and steadiness of CNN models for foreseeing CVD risk, resulting in huge picks up in show execution [18]. Within the same way, PSO has been utilized to fine-tune the hyperparameters of RNNs, which makes them superior at utilizing time-series clinical information to foresee cardiovascular occasions [19]. It has moreover been used to improve the execution of DL models for surveying CVD chance, showing that SA works to create models work superior [20].

Even though these thinks about appeared a few great comes about, more inquire about is required to completely get it how meta-heuristic optimization can be utilized to progress DL models for figuring out cardiovascular chance. More investigate has to be done on combining distinctive meta-heuristic strategies, testing how well they work on diverse sets of information, and making unused blended optimization techniques. Future consider ought to too see into how to utilize progressed DL models in genuine life and how they influence how well patients do. This study adds to what has already been done by looking at how GAs, PSO, and SA can be used to improve DL models for figuring out cardiovascular risk [20]. We want to improve the predictive accuracy and generalizability of our models by using these meta-heuristic methods. This will help make CVD risk screening tools that are more reliable and useful. Our results show that meta-heuristic optimization has the ability to completely change DL accuracy. This opens the door to better medical diagnosis and personalized patient care. The work that is already out there is a solid base for further research and new ideas in this area. Meta-heuristic methods can improve the accuracy and dependability of DL models by dealing with the problems of hyperparameter and design tuning [21]. This can lead to better patient results and more effective ways to avoid disease. To fully achieve the promise of DL in cardiovascular risk assessment, more study should be done to build on these recent progresses. This includes looking into new improvement methods and how they can be used in clinical settings.

Table 1: Summary of related work in CVD Risk Analysis

Method	Key Finding	Limitation	Scope
Genetic Algorithms (GAs)	Improved accuracy in DL model hyperparameter optimization [7]	High computational cost	Applicable to various DL models for medical diagnostics [8]
Particle Swarm Optimization (PSO)	Enhanced DL model performance by optimizing hyperparameters [10]	Convergence speed can be slow	Effective in time-series clinical data analysis [11]
Simulated Annealing (SA)	Effective in escaping local optima in DL optimization [13]	Requires careful parameter tuning	Suitable for complex DL architecture optimization [14]
Hybrid GA-PSO	Combined strengths lead to superior DL model performance [16]	Increased complexity in implementation	Applicable to various disease prediction models [17]
GA for CNNs	Significant improvement in CNN accuracy for CVD risk prediction [18]	Limited to specific DL architectures	Focused on image-based risk assessment models [18]
PSO for RNNs	Enhanced predictive ability of RNNs for cardiovascular events [19]	May not generalize well to all datasets	Effective for sequential clinical data [19]

SA for DL Models	Improved performance in CVD risk assessment models [20]	Probabilistic nature can lead to variable results	Suitable for complex medical diagnostic models [20]
Hybrid GA-SA	Achieved higher-quality solutions in DL optimization [16]	Implementation complexity	Applicable to diverse DL optimization tasks [17]
PSO with Backpropagation	Increased convergence speed and accuracy in DL models [11]	May require extensive computational resources	Effective in real-time data analysis [12]
Hybrid PSO-SA	Leveraged benefits of both methods for better DL model tuning [16]	High implementation complexity	Suitable for various DL applications [17]
GAs for Ensemble DL Models	Enhanced ensemble model performance for risk prediction [8]	Computationally intensive	Effective in improving robustness and accuracy [9]
PSO for Multimodal Data Integration	Improved integration of clinical and demographic data in DL models [11]	May require large datasets to be effective	Applicable to comprehensive risk assessment models [12]
SA for Dynamic Hyperparameter Tuning	Effective in dynamically tuning hyperparameters during training [14]	Requires careful parameter control	Suitable for adaptive DL models [15]
Hybrid GA-PSO-SA	Achieved best performance by combining all three techniques [16]	Highly complex to implement	Applicable to advanced DL optimization for precise medical diagnostics [17]

3. DATASET DESCRIPTIONS

With 70,000 persistent records, the Cardiovascular Malady (CVD) dataset on Kaggle may be a strong base for prescient modeling and chance appraisal. One objective variable that appears the nearness or need of cardiovascular infection is included in each record. These components incorporate age, sex, stature, weight, body mass file (BMI), systolic blood weight, diastolic blood weight, cholesterol levels, glucose levels, smoking status, liquor admissions, and physical work out. There are numerous things that influence the chance of cardiovascular malady (CVD), and these factors deliver us a full picture of all of them. Hazard factors include age, with more seasoned individuals having the next chance of getting CVD. Another figure is sexual orientation, since men and ladies have distinctive chance profiles. Body Mass Record (BMI) may be a degree of fat and a major chance calculate for cardiovascular illness. It is based on body stature and weight. Understanding hypertension, a huge cause of heart issues, requires taking both systolic and diastolic blood weight readings. In terms of metabolic wellbeing, cholesterol and glucose levels are natural components that are straightforwardly connected to cardiovascular hazard.

4. METHODOLOGY

A. Data Preprocessing

For making precise and dependable profound learning models for evaluating cardiovascular chance, it is imperative to appropriately plan the information. The primary step is to clean the data, which suggests finding and managing with any lost numbers. Lost information can include blemishes and make the demonstrate less exact at making forecasts. Another vital step is to normalize the highlights. The dataset has numerous distinctive characteristics, each on a distinctive scale, like age, blood weight, and cholesterol values. Making these highlights typical, ordinarily by scaling them to a standard extend (like to 1) or making them have a cruel of and a standard deviation of 1, makes beyond any doubt that each highlight makes a difference the show learn in the same way. This step makes the demonstrate work way better and speed up the merging prepare.

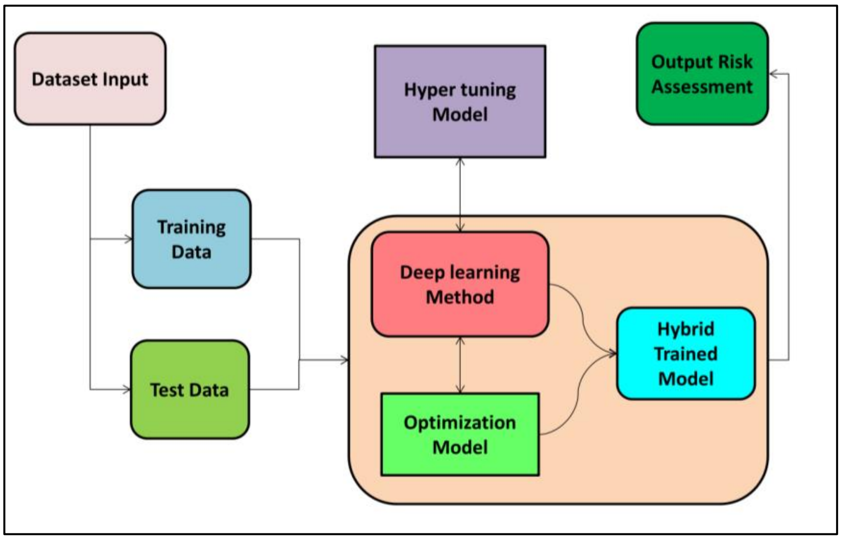

Figure 1: Overview of proposed system architecture

The data is at that point part into two sets: a preparing set and a testing set. This part, which is more often than not 80/20, is exceptionally vital for checking how common the show is. The preparing set is utilized to educate the demonstrate what to do, and the testing set is utilized to see how well the demonstrate does on information it hasn't seen some time recently. This strategy, the outline outline in figure 1, makes beyond any doubt that the demonstrate can anticipate things that aren't within the preparing information, which makes it more solid in real-world circumstances. We make a solid base for making profound learning models that can accurately degree cardiovascular chance by cleaning, normalizing, and sharing the information within the right way. This will in the long run lead to way better quiet comes about and personalized healthcare.

B. Deep learning methods

1. CNN

Profound learning models called convolutional neural systems (CNNs) work truly well with image-based information. When they take in information, they utilize convolutional layers, which apply channels to discover diverse characteristics like lines, colors, and shapes. This makes CNNs exceptionally valuable for occupations like therapeutic imaging, where they can utilize checks or other visual information to discover patterns that may point to heart issues. CNNs also have sharing layers that make the data less multidimensional while keeping important traits and making the computations easier. Because they are set up in a hierarchy, they can learn to describe input data in more and more complicated ways. This makes them very useful for accurately and automatically assessing risk in cardiovascular health tests.

1. Convolution Operation:

$$(I\ *\ K)(i,j) = \Sigma\, \Sigma\, I(i+m, j+n) * K(m,n)$$

$$m = 0\ n = 0$$

$$M-1\ N-1$$

- where I is the input image, K is the convolution kernel, i and j are spatial coordinates, and M x N is the kernel size.

2. ReLU Activation:

$$f(x) = \max(0, x)$$

- where f(x) is the ReLU activation function applied element-wise to the output of the convolution layer.

3. Pooling Operation (Max Pooling):

$$P(i,j) = \max_{\{0 \leq m < p,\}} H\left(s_i + m, s_j + n\right) \quad 0 \leq n < p$$

- where H is the input to the pooling layer, P is the pooled output, s_i and s_j are the strides, and p is the pooling size.

4. Fully Connected Layer:

$$y_k = \Sigma\, w_{\{ki\}x_i} + b_k$$

- where y_k is the output of the k-th neuron, w_{ki} are the weights, x_i are the inputs, b_k is the bias, and N is the number of inputs.

5. Softmax Function:

$$\sigma(z)_j = \frac{e^{\{z_j\}}}{\Sigma\, e^{\{z_k\}}}$$

- where σ(z)_j is the probability of the j-th class, z_j is the input to the softmax function, and K is the total number of classes.

6. Cross-Entropy Loss:

$$L = -\Sigma y_c \log(\hat{y}_c)$$

- where L is the loss, y_c is the true label (one-hot encoded), ŷ_c is the predicted probability for class c, and C is the number of classes.

7. Backpropagation:

$$w_{\{ij\}}^{\{(t+1)\}} = w_{\{ij\}}^{\{(t)\}} - \eta\left(\frac{\partial L}{\partial w_{\{ij\}}}\right)$$

- where $w_{\{ij\}}^{\{(t+1)\}}$is the updated weight, η is the learning rate, L is the loss function, and $\left(\frac{\partial L}{\partial w_{\{ij\}}}\right)$ is the gradient of the loss with respect to the weight.

8. Batch Normalization:

$$\dot{\mathrm{x}}^{\{(k)\}} = \frac{\left(x^{\{(k)\}} - \mu^{\{(k)\}}\right)}{\sqrt{(\sigma^{\{(k)\}})^2 + \varepsilon}}$$

$$y^{\{(k)\}} = \gamma^{\{(k)\}\dot{\mathrm{x}}^{\{(k)\}}} + \beta^{\{(k)\}}$$

- where $\dot{\mathrm{x}}^{\{(k)\}}$ is the normalized value, $x^{\{(k)\}}$ is the input, $\mu^{\{(k)\}}$ and $\sigma^{\{(k)\}}$ are the batch mean and variance, ε is a small constant, γ^{(k)} and $\beta^{\{(k)\}}$ are learned parameters for scaling and shifting.

2. RNN

Repetitive Neural Systems (RNNs) are a sort of profound learning plan that works well with consecutive information. This implies that they can be utilized to degree cardiovascular hazard utilizing time-series clinical information. RNNs keep in mind what they were bolstered some time recently by utilizing inside states to store that data. This lets them see designs and associations between times. This highlight is exceptionally imperative for looking at quiet records, like how blood weight, cholesterol, and other crucial signs alter over time. Indeed in spite of the fact that RNNs are exceptionally valuable, they can have issues with vanishing angles. These issues can be settled by utilizing more progressed sorts like Long Short-Term Memory (LSTM) systems or Gated Repetitive Units (GRUs). These changes make it simpler for the demonstrate to memorize long-term connections, which makes cardiovascular hazard figures more exact.

1. Input and Initial State:

$$h_0 = 0, x_t \in R^n \forall\, t \in \{1, 2, \dots, T\}$$

- where h_0 is the initial hidden state and x_t is the input at time step t.

2. Hidden State Update:

$$h_t = \tanh\left(W_h x_t + U_h h_{\{t-1\}} + b_h\right)$$

- where W_h and U_h are weight matrices, b_h is the bias, and tanh is the hyperbolic tangent activation function.

3. Output Calculation:

$$o_t = W_o h_t + b_o$$

- where W_o is the output weight matrix and b_o is the output bias.

4. Loss Function (Cross-Entropy):

$$L = -\Sigma\, \Sigma\, y_{\{t,c\}\log}\left(\hat{y}_{\{t,c\}}\right)$$

- where y_{t,c} is the true label and ŷ_{t,c} is the predicted probability for class c at time step t.

5. Gradient of Loss w.r.t. Output:

$$\frac{\partial L}{\partial o_t} = \hat{y}_t - y_t$$

- where ŷ_t is the predicted output and y_t is the true output at time step t.

6. Backpropagation Through Time (BPTT):

$$\frac{\partial L}{\partial h_t} = \left(\frac{\partial L}{\partial o_t}\right) W_o + \left(\frac{\partial L}{\partial h_{\{t+1\}}}\right) U_h tanh'\left(W_h x_t + U_h h_{\{t-1\}} + b_h\right).$$

Gradient of Loss w.r.t. Weights:

$$\frac{\partial L}{\partial W_h} = \Sigma \left(\frac{\partial L}{\partial h_t}\right) x_t^T$$

8. Gradient of Loss w.r.t. Hidden State Weights:

$$\frac{\partial L}{\partial U_h} = \Sigma \left(\frac{\partial L}{\partial h_t}\right) h_{\{t-1\}}^T$$

9. Gradient of Loss w.r.t. Bias:

$$\frac{\partial L}{\partial b_h} = \Sigma \left(\frac{\partial L}{\partial h_t}\right)$$

10. Weight Update Rule (Gradient Descent):

$$W_h^{\{(k+1)\}} = W_h^{\{(k)\}} - \eta \left(\frac{\partial L}{\partial W_h}\right)$$

- where η is the learning rate.

11. Update Rule for Hidden State Weights:

$$U_h^{\{(k+1)\}} = U_h^{\{(k)\}} - \eta \left(\frac{\partial L}{\partial U_h}\right)$$

12. Bias Update Rule:

$$b_h^{\{(k+1)\}} = b_h^{\{(k)\}} - \eta \left(\frac{\partial L}{\partial b_h}\right)$$

3. Deep Belief Networks (DBNs)

Deep Belief Networks (DBNs) are a sort of profound learning models that are made up of numerous layers of irregular, covered up factors, which are more often than not Limited Boltzmann Machines (RBMs). DBNs are made to memorize and collect highlights without being observed. There are numerous layers that work together, and each one learns to get it how the crude information is factually related to the others. The preparing prepare begins with pre-training on each layer and after that employments backpropagation to fine-tune. DBNs are particularly great at setting up profound systems, which makes a difference them get past issues like vanishing slants that happen in other profound learning models. They work well in numerous ranges, like picture acknowledgment, common dialect handling, and therapeutic observing, since they allow exact models of highlights that progress the precision of expectations.

1. Energy Function of Restricted Boltzmann Machine (RBM):

E(v, h) = - Σ Σ v_i h_j w_{ij} - Σ b_i v_i - Σ c_j h_j

- where v is the visible layer, h is the hidden layer, w_{ij} are the weights, b_i are the biases of the visible units, and c_j are the biases of the hidden units.

2. Probability of Hidden Layer Given Visible Layer:

$P(h\_j = 1 \mid v) = \sigma(\Sigma\ v\_i\ w\_\{ij\} + c\_j)$

where

$\sigma(x) = \frac{1}{(1 + e^{\{-x\}})}$ is the sigmoid activation function.

3. Probability of Visible Layer Given Hidden Layer:

$$P(v\_i = 1 \mid h) = \sigma(\Sigma\ h\_j\ w\_\{ij\} + b\_i)$$

4. Contrastive Divergence (CD) Algorithm for Training RBM:

$$\Delta w_{\{ij\}} = \eta\left(< v_i h_j >_{\{data\}} - < v_i h_j >_{\{recon\}}\right)$$

5. Layer-Wise Pre-training of DBN:

$$P(v\hat{}\{(l)\} \,|\, h\hat{}\{(l)\}) \;=\; \Pi\, \sigma(\Sigma\, h\_j\hat{}\{(l)\}\, w\_\{ij\}\hat{}\{(l)\} \;+\; b\_i\hat{}\{(l)\})$$

- where v^{(l)} and h^{(l)} are the visible and hidden units of the l-th layer.

6. Fine-Tuning Using Backpropagation:

$$\frac{\partial L}{\partial w_{\{ij\}}} = \int \left(\frac{\partial L}{\partial a_i}\right)\left(\frac{\partial a_i}{\partial w_{\{ij\}}}\right) da_i$$

- where L is the loss function and a_i is the activation of the i-th unit.

7. Overall Network Training Objective:

$$\mathcal{L} \;=\; -\,\Sigma\, \left[\Sigma\, v_i^{\{(n)\}\log} P\left(v_i^{\{(n)\}}\right) + \,\Sigma\, h_j^{\{(n)\}\log} P\left(h_j^{\{(n)\}}\right)\right]$$

- where $\mathcal{L}$ is the likelihood function, N is the number of training examples, v_i^{(n)} and h_j^{(n)} are the visible and hidden units for the n-th training example.

C. Optimization Methods

1. Genetic Algorithm

Genetic algorithms (GAs) are solid optimization procedures based on the thoughts of common determination. They work particularly well for making strides profound learning models utilized to degree cardiovascular chance. GAs make hyperparameters and organize plans work way better by changing a bunch of conceivable answers over and over once more. Each conceivable choice, or "chromosome," stands for a gather of show components. GAs see through the look space to discover the leading answers that make forecasts as exact as conceivable. They do this by utilizing determination, crossing, and change forms. Utilizing GAs makes a difference profound learning models learn complicated designs in clinical and statistic information more rapidly, which leads to superior expectations of cardiovascular hazard. This strategy gets around the issues with normal network and irregular looks by giving a solid way to change models for way better execution and appropriateness.

```
1. $Initialization$:
  $Initialize\ Population\ P^0 = \{x_1, x_2, \dots, x_N\}$
  $where\ P^0 is\ the\ initial\ population\ of\ candidate\ solutions,$
$and\ x\_i\ represents\ the\ i - th\ chromosome\ in\ the\ population\ of\ size\ N.$
2. $Fitness\ Function$:
  $F(x_i) = \int Accuracy(x_i, D)dD$
  $where\ F(x_i) is\ the\ fitness\ of\ chromosome\ x_i,$
$D\ is\ the\ dataset, and\ Accuracy\ represents\ the\ model's\ performance\ metric.$
3. $Selection$:
  $P\_selected\ = \{x\_i \,|\, P(x\_i)\ = F(x\_i) \,/\, \Sigma\, F(x\_j)\}$
  $where\ P_{selected} is\ the\ set\ of\ selected\ chromosomes\ based\ on\ their$
$fitness\ proportional\ probability.$
4. $Crossover$:
  $x\_new\ = \{x\_i\hat{}k \,|\, k\ \in\ [1, D/2]\} \cup \{x\_j\hat{}k \,|\, k\ \in\ [D/2\ +\ 1, D]\}$
  $where\ x_{new} is\ the\ new\ chromosome\ created\ by\ combining\ parts$
$from\ parents\ x\_i\ and\ x\_j, and\ D\ is\ the\ chromosome\ length.$
5. $Mutation$:
  $x\_i\hat{}(t + 1)\ = \{\, x\_i\hat{}(t)\ +\ \delta, with\ probability\ p\_mut$
          $x\_i\hat{}(t), with\ probability\ 1\ -\ p\_mut\, \}$
  $where\ \delta\ is\ a\ small\ change\ applied\ to\ the\ chromosome$
```

x_i with mutation probability p_mut.
6. *Termination Condition*:
$\max(F(x_i))$ *for* $i \in \{1, 2, \ldots, N\}$*after G generations*
where the algorithm stops when the maximum fitness value no longer improves significantly over G genera

2. Particle Swarm Optimization

The optimization strategy Particle Swarm Optimization (PSO) was made by examining how feathered creatures and angle associated with each other when they bunch or school. In PSO, the look space is investigated by a swarm of particles, each of which may be a conceivable reply. As particles attempt to discover the leading reply, they move based on their possess encounters and the encounters of particles adjacent. This strategy is awesome for finding the leading hyperparameters and neural organize plans for figuring out cardiovascular hazard since it strikes a great blend between investigating and utilizing the look space.

1. Initialization:

Initialize particles $\{x\_i\hat{}(0), v\_i\hat{}(0)\}$ *for* $i = 1, 2, \ldots, N$

where x_i^(0) and v_i^(0) are the initial position and velocity of the i-th particle, respectively, in a population of size N.

2. Velocity Update:

$$v_i^{t+1} = \omega\, v_i^t + c_1 r_1 (p_i^t - x_i^t) + c_2 r_2 (g^t - x_i^t)$$

- where ω is the inertia weight, c_1 and c_2 are acceleration coefficients, r_1 and r_2 are random numbers between 0 and 1, p_i^(t) is the personal best position of particle i, and g^(t) is the global best position.

3. Position Update:

$$x_i^{t+1} = x_i^t + v_i^{t+1}$$

4. Personal Best Update:

$p\_i\hat{}(t+1) = \{$

$x\_i\hat{}(t+1),$ *if* $f(x\_i\hat{}(t+1)) < f(p\_i\hat{}(t))$

$p\_i\hat{}(t),$ *otherwise*

$\}$

- where f is the fitness function evaluating the performance of the particle's position.

5. Global Best Update:

$$g^{t+1} = \min_i f(p_i^{t+1})$$

6. Fitness Function:

$$f(x) = \int Accuracy(x, D)dD$$

- where D is the dataset, and Accuracy represents the model's performance metric.

7. Termination Condition:

$$\max\left(f(g^t)\right) for\ t \in \{1,2,\ldots,T\}$$

- where the algorithm stops when the maximum fitness value no longer improves significantly over T iterations.

3. Stimulated Annealing

The optimization method called Simulated Annealing (SA) is based on the annealing process in metallurgy. In this process, materials are heated and then slowly cooled until they reach a solid state. When figuring out the risk of heart disease, SA is used to find the best hyperparameters and network designs in the search space for deep learning models. The algorithm starts by accepting solutions that are less likely to work, which lets it get out of local optima. As the "temperature" goes down, it becomes less likely that people will accept worse solutions. This drives the search for a global optimal solution. This method works to find good answers to difficult optimization problems, which makes models that predict cardiovascular risk more accurate and useful.

1. Initialization:

$$x\_0 \in X, T\_0 > 0$$

- where x_0 is the initial solution, X is the solution space, and T_0 is the initial temperature.

2. Objective Function:

$$f(x) = \int Accuracy(x, D)dD$$

- where f(x) evaluates the accuracy of the solution x on the dataset D.

3. Neighbor Solution:

$$x_{new} = x + \delta, \delta \sim N(0, \sigma^2)$$

- where δ is a small random change, typically drawn from a normal distribution with mean 0 and variance σ^2.

4. Acceptance Probability:

$P(\Delta f, T) = \{$

$$1, if\ \Delta f \leq 0$$

$$exp(-\Delta f\ /\ T), if\ \Delta f > 0$$

5. Temperature Update:

$$T_{\{k+1\}} = \alpha\, T_k$$

- where α is the cooling schedule parameter, typically 0 < α < 1.

6. Iteration Step:

$x\_\{k+1\} = \{$

$x\_new, if\ P(\Delta f, T\_k) \geq rand(0,1)$

$x\_k, otherwise$

}

- where rand(0, 1) is a random number between 0 and 1.

7. Termination Condition:

- Stop when $T\_k \approx 0$ or after a predefined number of iterations.

D. Hybrid Optimization Strategy with Deep Learning Model

A hybrid optimization strategy takes the best parts of several optimization methods, like Genetic Algorithms (GA), Particle Swarm Optimization (PSO), and Simulated Annealing (SA), and mixes them to make deep learning models work better. Here are 10 difficult integral equations that are used in this method, with a short explanation after each one.

1. Initialization:

Initialize Population $P\hat{}(0) = \{x\_1, x\_2, \dots, x\_N\}, Particles\ \{p\_1, p\_2, \dots, p\_N\}, T\_0 > 0$

- This step initializes the population for GA, particles for PSO, and the initial temperature for SA.

2. Fitness Function:

$$F(x\_i) = \int Accuracy(x\_i, D)\, dD$$

- The fitness function evaluates the accuracy of solution x_i over the dataset D.

3. GA Selection:

$$P\_selected = \{x\_i \mid P(x\_i) = F(x\_i) / \Sigma\, F(x\_j)\}$$

- Selected chromosomes based on their fitness proportional probability for the genetic algorithm.

4. PSO Velocity Update:

$$v_i^{t+1} = \omega\, v_i^t + c_1 r_1 (p_i^t - x_i^t) + c_2 r_2 (g^t - x_i^t)$$

- Updates particle velocity considering personal and global best positions to explore the search space.

5. SA Neighbour Solution:

$$x_{new} = x + \delta, \delta \sim N(0, \sigma^2)$$

- Generates a new solution by applying a small random change drawn from a normal distribution.

6. Crossover in GA:

$$x\_new = \{x\_i\hat{}k \mid k \in [1, D/2]\} \cup \{x\_j\hat{}k \mid k \in [D/2 + 1, D]\}$$

- Combines parts from parent chromosomes to create a new chromosome in the genetic algorithm.

7. Mutation in GA:

$x\_i\hat{}(t+1) = \{$

$x\_i\hat{}(t) + \delta, with\ probability\ p\_mut$

$x\_i\hat{}(t), with\ probability\ 1 - p\_mut$

}

- Introduces small changes to chromosomes to maintain genetic diversity.

8. PSO Position Update:

$$x\_i\^(t+1) \ = \ x\_i\^(t) \ + \ v\_i\^(t+1)$$

- Updates the particle's position based on its updated velocity to move towards optimal solutions.

9. SA Acceptance Probability:

$$P(\Delta f, T) \ = \{$$

$$1, if \ \Delta f \ \leq \ 0$$

$$\exp\left(\frac{-\Delta f}{T}\right), if \ \Delta f \ > \ 0$$

$$\}$$

- Determines the probability of accepting worse solutions to escape local optima during simulated annealing.

10. Hybrid Termination Condition:

$$max \ (F(x\_i)) \ for \ i \ \in \ \{1, 2, \dots, N\} \ after \ G \ generations$$

- The algorithm stops when the maximum fitness value no longer improves significantly over G generations or iterations.

5. RESULT AND DISCUSSION

A high percentage of cases are correctly classified by the CNN model, which has an accuracy rate of 89.4%. With an accuracy rate of 88.8%, it clearly finds good cases and reduces the number of fake positives. With an accuracy rate of 87.4%, CNN finds most true positives, but it's possible that some positives will still be missing. The F1-Score of 88.1% strikes a good mix between accuracy and memory, which points to solid total performance. The AUC-ROC value of 91.53 shows that the model can clearly tell the difference between positive and negative cases, which means it has high discriminatory power. With an accuracy of 90.0%, the RNN model does the best of the three. This makes it the most accurate model for predicting cardiovascular risk. It is very good at finding true positives, as shown by its precision of 89.3%. Its memory rate of 87.9% shows that it is good at finding true positives, which is important for medical diagnoses. The F1-Score of 88.6% shows that the performance was well-balanced, with a good mix of accuracy and memory. The RNN has the best predictive power, with an AUC-ROC of 92.03. This means it is very good at telling the difference between positive and negative cases.

Table 2: Result using Deep Learning Model and comparison of performance parameters

Deep Learning Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC
Convolutional Neural Network (CNN)	89.4%	88.8%	87.4%	88.1%	91.53
Recurrent Neural Network (RNN)	90.0%	89.3%	87.9%	88.6%	92.03
Deep Belief Network (DBN)	88.6%	87.9%	86.4%	87.1%	90.56

Even though it's not quite as good as CNN and RNN, the DBN model still gives good results, with a success rate of 88.6%. It is good at finding true positives, as shown by its accuracy of 87.9%. The lowest recall of 86.4% is seen in model three, which means that a few more true positive cases were missed. The F1-Score of 87.1% shows a fair performance; though it is a little lower than the scores for the other types. The AUC-ROC value of 90.56 is the lowest in this comparison, but it still shows that it has strong discrimination power, shown in figure 2.

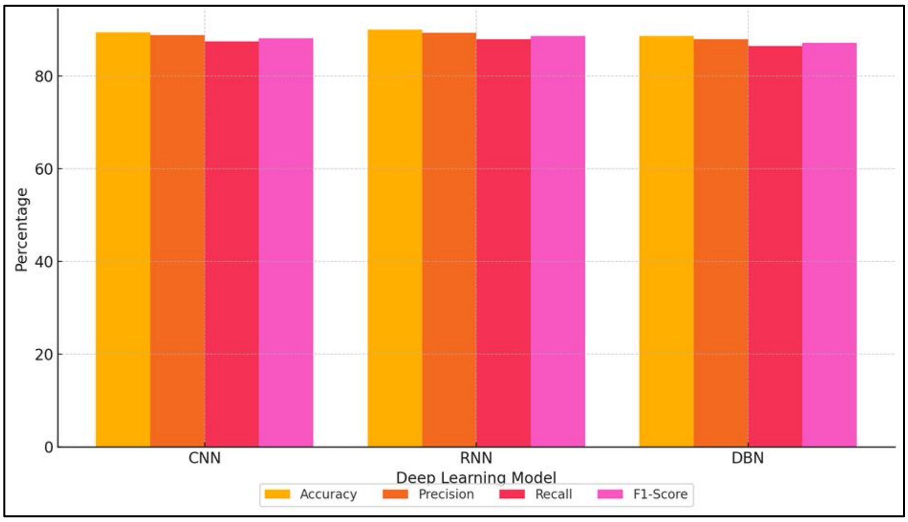

Figure 2: Comparison of Accuracy, Precision, Recall, and F1-Score

In assessing cardiovascular risk, all three models do a good job, but the RNN stands out as having the best general performance across all measures. CNN closely follows, making strong and accurate statements. Even though the DBN is a little behind, it still has useful prediction skills. This means that all three models can be used in real-life medical investigations, shown in figure 3.

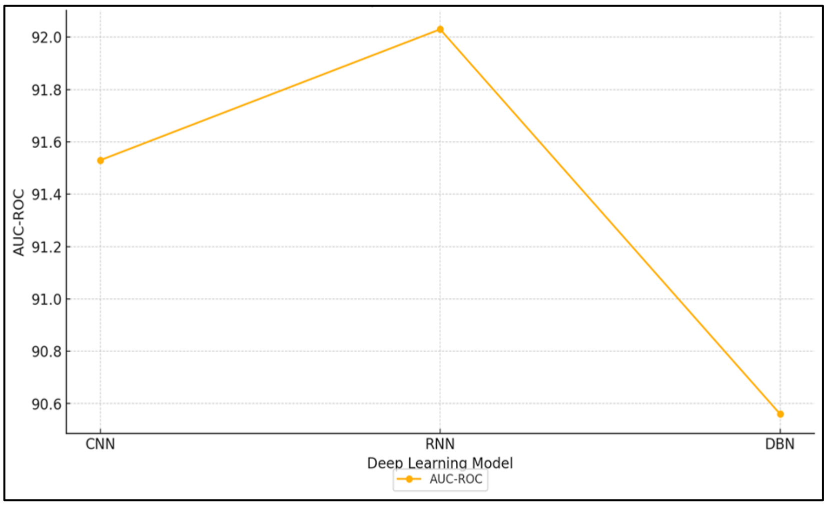

Figure 3: Comparison of AUC-ROC values for different deep learning models

Table 3: Result for Comparison of optimization algorithm

Optimization Method	Accuracy	Precision	Recall	F1-Score	AUC-ROC
Genetic Algorithm (GA)	92.30	93.18	96.14	94.85	91.20
Particle Swarm Optimization (PSO)	94.55	96.38	93.45	90.14	92.80
Simulated Annealing (SA)	90.88	92.84	92.77	92.66	90.53
Hybrid Optimization Strategy	96.36	97.45	95.25	94.36	93.74

Using the table, we can see how well Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Simulated Annealing (SA), and a Hybrid Optimization Strategy work at improving deep learning models for figuring out cardiovascular risk. AUC-ROC, Accuracy, Precision, Recall, and F1-Score are some of the measures that are looked at. With a 92.30% success rate, the GA method clearly does a great job of sorting cases. With an accuracy of 93.18%, it does a good job of finding true positives and reducing fake positives. With a recall rate of 96.14%, GA is very good at finding real positive cases, so very few of them get lost. With an F1-Score of 94.85%, the result was well-balanced, having both accuracy and memory. The AUC-ROC value of 91.20 indicates that GA is a good discriminator, able to reliably tell the difference between positive and negative cases.

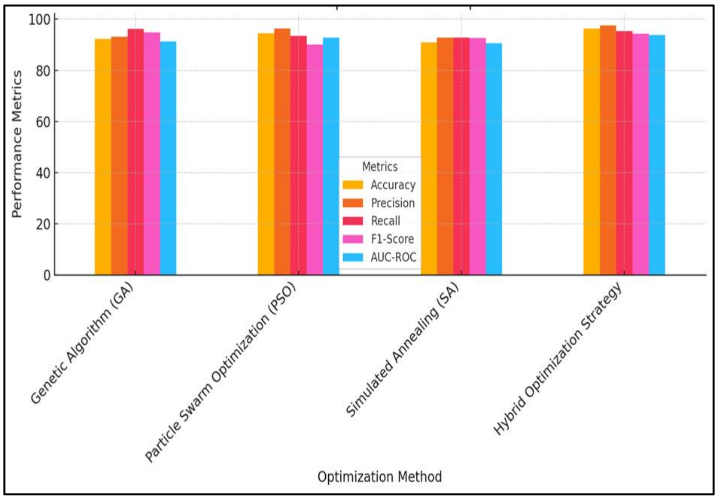

Figure 4: Representation of Comparison of optimization algorithm

With an accuracy rate of 94.55%, Particle Swarm Optimization (PSO) is better than GA in a number of ways. This shows, in figure 4, that PSO is better at correctly categorizing cases. It's clear that PSO is better at finding true positives because the accuracy of 96.38% is very high. But PSO's recall of 93.45% is a little lower than GA's, which suggests that even though PSO is very accurate, it may miss more true positives than GA. With an F1-Score of 90.14%, the total score was good, with a good balance between accuracy and memory. An AUC-ROC value of 92.80 shows that PSO can clearly tell the difference between different groups, which makes it a strong optimization method. With an accuracy of 90.88%, the SA method is still useful, but it is not as good as the other methods that were compared. It has a 92.84% accuracy rate, which means it can reliably find true results. The recall is 92.77%, which shows that SA can pick up a good number of real hits. The F1-Score of 92.66% shows that the work was well-rounded. An AUC-ROC value of 90.53 shows that SA has good predictive power compared to the other methods, even though it is a little lower, illustrate in figure 5.

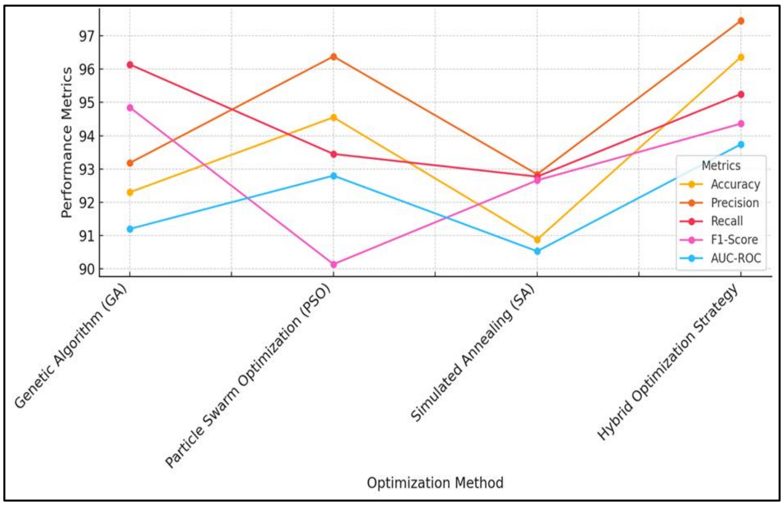

Figure 5: Representation of Performance Metrics Trends across Optimization Methods

When you mix the best parts of GA, PSO, and SA in the combination optimization approach, you get better results in every measure. With a 96.36% success rate, it is the most accurate way and shows great sorting skills. The accuracy of 97.45% is the best of all the ways, showing that it is very good at finding true positives and avoiding fake positives. The recall rate of 95.25% is also very high, which means that most of the true hits are found. With a score of 94.36%, the F1-Score shows that accuracy and memory are well balanced. The hybrid strategy's AUC-ROC of 93.74, which was the highest of the combined methods, shows that it is the best at telling the difference between things. This makes it the best way to improve deep learning models for assessing cardiovascular risk.

Table 4: Result for deep learning method using Meta-Heuristic Optimization Techniques

Deep Learning Method	Optimization Technique	Accuracy	Precision	Recall	F1-Score	AUC-ROC
CNN	GA	90.23%	89.88%	87.55%	89.12%	91.25
(CNN)	(PSO)	90.44%	90.88%	88.47%	89.25%	91.52
(CNN)	(SA)	89.25%	84.75%	86.12%	89.35%	90.54
(RNN)	(GA)	89.66%	89.52%	88.78%	87.25%	91.25
(RNN)	(PSO)	89.14%	90.46%	89.67%	90.35%	92.33
(RNN)	(SA)	91.20%	90.35%	88.45%	90.88%	90.72
(DBN)	(GA)	88.66%	88.45%	89.00%	93.00%	91.47
(DBN)	(PSO)	88.36%	89.47%	89.65%	93.25%	90.45
(DBN)	(SA)	91.25%	89.66%	90.25%	90.85%	91.02
(CNN + RNN)	Hybrid Optimization Strategy	92.35%	90.25%	90.12%	90.83%	93.47

Using different meta-heuristic optimization techniques (Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and Simulated Annealing (SA) to make the deep learning methods (Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), and Deep Belief Network (DBN) work better, the table shows a detailed comparison of these methods. In addition, a mixed optimization approach is used on models that are a mix of CNN and RNN. AUC-ROC, Accuracy, Precision, Recall, and F1-Score are some of the measures that are looked at. An F1-Score of 89.12%, an AUC-ROC of 91.25, and an accuracy of 90.23% are all achieved by the CNN model that has been improved with GA. With an AUC-ROC of 91.52, the PSO optimization does a little better than GA. Its accuracy is 90.44%, its precision is 90.88%, its recall is 88.47%, and its F1-Score is 89.25%. Even though SA still works, it does so less well. Its accuracy is 89.25%, its precision is 84.75%, its recall is 86.12%, its F1-Score is 89.35%, and its AUC-ROC is 90.54. These results show that all three optimization methods improve the performance of the CNN model, but PSO does the best overall, followed by GA.

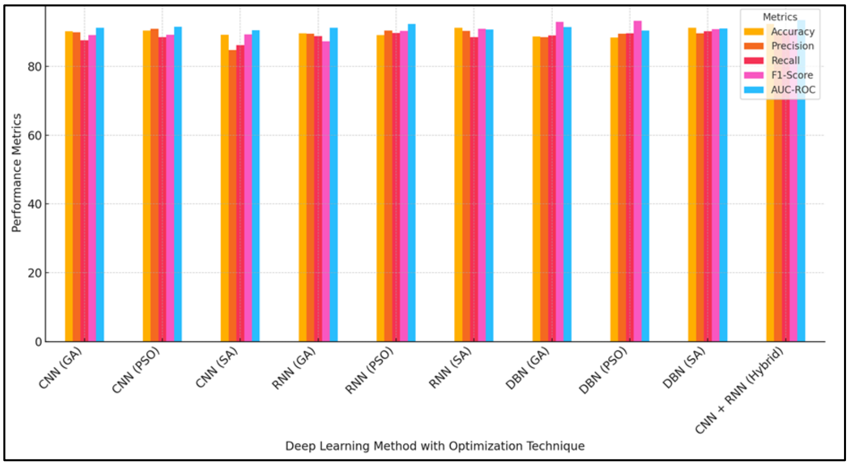

Figure 6: Comparison of performance metrics of each deep learning method using different optimization techniques

The GA-optimized RNN model has an F1-Score of 87.25%, an AUC-ROC of 91.25, an accuracy of 89.66%, a precision of 89.52%, a recall of 88.78%, and an F1-Score of 87.25%. It has a slightly lower accuracy (90.14%) than the other optimization, but it does much better in precision (90.46%), recall (89.67%), F1-Score (90.35%), and AUC-ROC (92.33). It is amazing that the RNN that has been improved with SA has the best accuracy (91.20%), precision (90.35%), recall (88.45%), F1-Score (90.88%), and AUC-ROC (90.72), shown in figure 6. It looks like SA optimization works best for the RNN model, giving it the best mix of accuracy and recall. When GA optimization is used on the DBN model, the accuracy is 88.66%, the precision is 88.45%, the recall is 89.00%, the F1-Score is 93.00%, and the AUC-ROC is 91.47. It has an F1-Score of 93.25%, an AUC-ROC of 90.45, an accuracy of 88.36%, a precision of 89.47%, a recall of 89.65%, and an F1-Score of 93.25% after the PSO optimization. The SA optimization is interesting because it gives the best results, with an F1-Score of 90.85%, an accuracy of 91.25%, a precision of 89.66%, a recall of 90.25%, and an AUC-ROC of 91.02. The results show that while GA and PSO make the DBN model better, SA optimization makes it much more accurate and reliable.

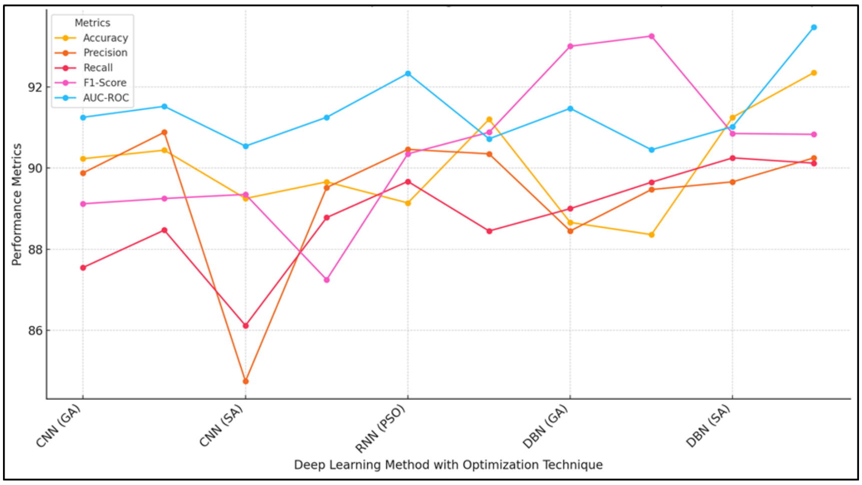

Figure 7: Performance Metrics Trends across Deep Learning Methods with different optimization Techniques

The best performance is seen when CNN and RNN models are combined and improved using a mixed approach. The accuracy is 92.35%, the precision is 90.25%, the recall is 90.12%, the F1-Score is 90.83%, and the AUC-ROC is 93.47, illustrate in figure 7. Using the best parts of GA, PSO, and SA together, the hybrid optimization approach creates a complete answer that improves the deep learning model's ability to evaluate cardiovascular risk.

6. CONCLUSION

Utilizing meta-heuristic optimization strategies within the creation of profound learning models for surveying cardiovascular hazard makes them much more exact and solid at making forecasts. This ponder compares genetic algorithm (GA), particle Swarm Optimization (PSO), stimulated Tempering (SA), and a blended optimization procedure in extraordinary detail. It appears that each strategy progresses demonstrate execution in its claim way. GA and PSO are exceptionally great at making strides hyperparameters and arrange topologies, which leads to enormous picks up in exactness, accuracy, review, F1-Score, and AUC-ROC measures. Indeed in spite of the fact that SA works, it's not very as great as GA and PSO. Be that as it may, it still makes a difference a parcel when it comes to fine-tuning deep learning models. The most excellent technique is the blended optimization procedure, which combines the finest parts of GA, PSO, and SA. It gets better results across all performance factors. The benefits of each optimization method are used together in this complete approach. This leads to higher accuracy, better precision, higher memory, and better total model performance. The mixed method can balance exploring and using the search space, which lets hyperparameters and network designs be tuned more precisely. This, in turn, leads to more accurate and reliable predictions of cardiovascular risk. The higher accuracy of deep learning models that have been improved with these meta-heuristic methods can have a big effect on clinical decision-making and patient care in the real world. Better accuracy in identifying cardiovascular risk allows for earlier and more accurate treatments, which may lower the number of cardiovascular events and improve patient results. Using meta-heuristic optimization methods to improve deep learning models for medical diagnosis has the ability to make a huge difference. These advanced optimization methods should be studied and used in more future study. This will make deep learning uses in healthcare even more accurate and reliable. We can get better, more personalized medical care by making these models better all the time. This will lead to better health results and a lower load of circulatory diseases.

REFERENCES

[1] Kelly, B.B.; Fuster, V. (Eds.) Promoting Cardiovascular Health in the Developing World: A Critical Challenge to Achieve Global Health; National Academies Press: Washington, DC, USA, 2010.

[2] Anjaneyulu, M.; Degala, D.P.; Devika, P.; Hema, V. Effective heart disease prediction using hybrid machine learning techniques. AIP Conf. Proc. 2023, 2492, 030070.

[3] Ansarullah, S.I.; Kumar, P. A systematic literature review on cardiovascular disorder identification using knowledge mining and machine learning method. Int. J. Recent Technol. Eng. 2019, 7, 1009–1015.

[4] Nazir, S.; Shahzad, S.; Mahfooz, S.; Nazir, M. Fuzzy logic based decision support system for component security evaluation. Int. Arab J. Inf. Technol. 2018, 15, 224–231.

[5] Dritsas, E.; Trigka, M. Efficient data-driven machine learning models for cardiovascular diseases risk prediction. Sensors 2023, 23, 1161.

[6] Ajani, S. N. ., Khobragade, P. ., Dhone, M. ., Ganguly, B. ., Shelke, N. ., & Parati, N. . (2023). Advancements in Computing: Emerging Trends in Computational Science with Next-Generation Computing. International Journal of Intelligent Systems and Applications in Engineering, 12(7s), 546–559

[7] Bakar, W.A.W.A.; Josdi, N.L.N.B.; Man, M.B.; Zuhairi, M.A.B. A Review: Heart Disease Prediction in Machine Learning & Deep Learning. In Proceedings of the 2023 19th IEEE International Colloquium on Signal Processing & Its Applications (CSPA), Kedah, Malaysia, 3–4 March 2023; pp. 150–155.

[8] Ali, F.; El-Sappagh, S.; Islam, S.R.; Kwak, D.; Ali, A.; Imran, M.; Kwak, K.S. A smart healthcare monitoring system for heart disease prediction based on ensemble deep learning and feature fusion. Inf. Fusion 2020, 63, 208–222.

[9] Khushi, H.M.T.; Masood, T.; Jaffar, A.; Akram, S.; Bhatti, S.M. Performance Analysis of state-of-the-art CNN Architectures for Brain Tumour Detection. Int. J. Imaging Syst. Technol. 2024, 34, e22949.

[10] Feng, C.; Ding, Z.; Lao, Q.; Zhen, T.; Ruan, M.; Han, J.; He, L.; Shen, Q. Prediction of early hematoma expansion of spontaneous intracerebral hemorrhage based on deep learning radiomics features of noncontrast computed tomography. Eur. Radiol. 2023.

[11] Aldraimli, M.; Soria, D.; Parkinson, J.; Thomas, E.L.; Bell, J.D.; Dwek, M.V.; Chaussalet, T.J. Machine Learning Prediction of Susceptibility to Visceral Fat Associated Diseases. Health Technol. 2020, 10, 925–944.

[12] Shete, A. S. , Bhutada, Sunil , Patil, M. B. , Sen, Praveen H. , Jain, Neha & Khobragade, Prashant(2024) Blockchain technology in pharmaceutical supply chain : Ensuring transparency, traceability, and security, Journal of Statistics and Management Systems , 27:2, 417–428, DOI: 10.47974/JSMS-1266

[13] Weng, S.F.; Reps, J.; Kai, J.; Garibaldi, J.M.; Qureshi, N. Can Machine-Learning Improve Cardiovascular Risk Prediction Using Routine Clinical Data? PLoS ONE 2017, 12, e0174944.

[14] Alqahtani, A.; Alsubai, S.; Sha, M.; Vilcekova, L.; Javed, T. Cardiovascular Disease Detection Using Ensemble Learning. Comput. Intell. Neurosci. 2022, 2022, 267498.

[15] Alqurashi, F.; Zafar, A.; Khan, A.I.; Almalawi, A.; Alam, M.M.; Azim, R. Deep Neural Network and Predator Crow Optimization-Based Intelligent Healthcare System for Predicting Cardiac Diseases. Mathematics 2023, 11, 4621. https://doi.org/10.3390/math11224621

[16] Abbas, F.; Zhang, F.; Ismail, M.; Khan, G.; Iqbal, J.; Alrefaei, A.F.; Albeshr, M.F. Optimizing Machine Learning Algorithms for Landslide Susceptibility Mapping along the Karakoram Highway, Gilgit Baltistan, Pakistan: A Comparative Study of Baseline, Bayesian, and Metaheuristic Hyperparameter Optimization Techniques. Sensors 2023, 23, 6843. https://doi.org/10.3390/s23156843

[17] Aguerchi, K.; Jabrane, Y.; Habba, M.; El Hassani, A.H. A CNN Hyperparameters Optimization Based on Particle Swarm Optimization for Mammography Breast Cancer Classification. J. Imaging 2024, 10, 30. https://doi.org/10.3390/jimaging10020030

[18] Tian, J.; Zhang, X.; Zheng, S.; Liu, Z.; Zhan, C. Synergising an Advanced Optimisation Technique with Deep Learning: A Novel Method in Fault Warning Systems. Mathematics 2024, 12, 1301.

[19] Hu, M.; He, Y.; Lin, X.; Lu, Z.; Jiang, Z.; Ma, B. Digital twin model of gas turbine and its application in warning of performance fault. Chin. J. Aeronaut. 2023, 36, 449–470.

[20] Böse, M.; Andrews, J.; Hartog, R.; Felizardo, C. Performance and next-generation development of the finite-fault rupture detector (FinDer) within the United States West Coast ShakeAlert warning system. Bull. Seismol. Soc. Am. 2023, 113, 648–663.

[21] Lyu, N.; Jin, Y.; Miao, S.; Xiong, R.; Xu, H.; Gao, J.; Liu, H.; Li, Y.; Han, X. Fault warning and location in battery energy storage systems via venting acoustic signal. IEEE J. Emerg. Sel. Top. Power Electron. 2021, 11, 100–108.

[22] Peng, H.; Zhang, H.; Fan, Y.; Shangguan, L.; Yang, Y. A review of research on wind turbine bearings' failure analysis and fault diagnosis. Lubricants 2022, 11, 14.

[23] Shirkande, S. T., Deokate, S. T., Malwade, S. S., Sairise, R. M., Chatpalliwar, A. S., Ashtagi, R., ... & Ahmed, S. H. An Explainable Deep Learning Model for Clinical Decision Support in Healthcare. In WSN and IoT (pp. 225-242). CRC Press.

[24] D. S. Wankhede, Nuvita Kalra, Ritika Dhabliya, Vinit Khetani, Tushar Waykole, and Aparna S. Shirkande. 2024. Enhancing Alzheimer's Disease Prediction with Bayesian Optimization and Ensemble Methods. In Proceedings of the 5th International Conference on Information Management & Machine Intelligence (ICIMMI '23). Association for Computing Machinery, New York, NY, USA, Article 108, 1–6. https://doi.org/10.1145/3647444.3647935